PLANEJAMENTO DE HORÁRIOS DE AULA EM UNIVERSIDADES

Claudio Luis de Amorim

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS (M.Sc.).

Aprovada por:

Fernando Chiyoshi

Prof.Fernando Yassuo Chiyoshi
Presidente

Nelson Maculan Fo

Prof. Nelson Maculan Filho

Ruderico F Pimentel

Prof. Ruderico Ferraz Pimentel

RIO DE JANEIRO, RJ - BRASIL
MAIO DE 1979

AMORIM, CLAUDIO LUIS DE
Planejamento de horários de aula em Universidades
[Rio de Janeiro] 1979.
VIII, 85 p. 29,7 cm (COPPE - UFRJ, M.Sc, Engenharia de Sistemas, 1979).

Tese - Universidade Federal do Rio de Janeiro.

1. Planejamento de horários de aulas I. COPPE/UFRJ. II, Título (série).

. A

ANA

AGRADECIMENTOS

Ao professor Fernando Y. Chiyoshi pela orientação valiosa e pelo princípio de liberdade com que sempre pautou o relacionamento com seu orientado.

Ao Dr. Luiz Carlos Galvão Lobo, Diretor do Núcleo de Tecnologia Educacional para a Saúde (NUTES/UFRJ) pelo estímulo e apoio à realização deste estudo.

A Ida M. L. Teixeira e Marisa M. Albuquerque pela orientação e a Iraídes M. P. Coelho, Alzira Salles e Jonathas Costa pelo trabalho realizado na elaboração do original desta tese.

"Não tenho compromissos com minhas idéias, busco a verdade".

Anisio Teixeira

R E S U M O

O planejamento de horários de aula em Universidades é formulado considerando uma multiplicidade de objetivos a serem atingidos, tais como:

1. A otimização dos recursos de salas existentes.
2. A uniformidade da carga horária dos cursos
3. A minimização dos horários vagos das turmas.
4. Satisfazer as prioridades relativas aos horários das aulas.

sujeito à disponibilidade semanal das salas, professores, turmas e horários.

Em face desta variedade de objetivos é proposta uma estratégia de solução e na qual são baseados e desenvolvidos procedimentos heurísticos de modo que os resultados obtidos pudessem ser aplicados a problemas de planejamento de horários reais.

A B S T R A C T

The planning of University timetables is formulated by considering the multiplicity of objectives to be reached such as:

1. Optimization of the avaiable classroom resources.
2. Uniformity of work-load of each course.
3. Minimization of waiting time between classes.
4. To satisfy priorities related to class timetables.

restrict to the weekly avaiability of classrooms, teachers, classes and time.

On considering this variety of objectives a strategy of solution is proposed in which heuristic procedures are based and developedin such a way that the results obtained might be applied to practical timetable problems.

ÍNDICE

INTRODUÇÃO

CAPÍTULO I - O PLANEJAMENTO DE HORÁRIOS DE AULA

1.1 - Apresentação do problema 3
1.2 - Revisão dos trabalhos desenvolvidos 4
1.3 - Considerações sobre os trabalhos 17
1.4 - Estratégia de solução adotada 18

CAPÍTULO II - PROGRAMAÇÃO DAS AULAS NOS DIAS DA SEMANA

2.1 - Introdução 22
2.2 - Formulação Matemática 23
2.3 - Descrição do Algorítmo 27
2.4 - Aplicação 29
2.5 - Testes e resultados 39
2.6 - Resumo do programa 42

2.6.1 - Rotina: PRINCIPAL 43
2.6.2 - Rotina: ALTERNATIVAS 44
2.6.3 - Rotina: SOLUÇÃO 44
2.6.4 - Rotina: GERA 45
2.6.5 - Rotina: ORDENA 46

CAPÍTULO III - PROGRAMAÇÃO DE HORÁRIOS DAS AULAS

3.1 - Introdução 47

3.2 - Formulação Matemática 48

3.3 - Descrição do Algorítmo 51

3.4 - Testes e Resultados 61

3.5 - Implementação do Algorítmo 63

3.5.1 - Rotina: PRINCIPAL 63

3.5.2 - Rotina: PARTIÇÃO 64

3.5.3 - Rotina: OCUPAÇÃO 65

3.5.4 - Rotina: TEMPO VAGO 66

CAPÍTULO IV - CONCLUSÕES E RECOMENDAÇÕES PARA FUTUROS TRABALHOS

4.1 - Conclusões e Recomendações 71

APÊNDICE 1 73

1.1 - Teste do algorítmo para a programação de horários nos dias da semana. 74

1.2 - Matrizes curso x sala segundo os dias da semana 78

1.3 - Parâmetros das rotinas 80

APÊNDICE 2 - PROGRAMAÇÃO DE HORÁRIOS PARA O TESTE EM 1.1 82

BIBLIOGRAFIA 84

INTRODUÇÃO

Com o número crescente de alunos tendo acesso às Universidades, estas se vêem cada vez mais compelidas a replanejarem suas atividades de ensino, pesquisa e administrativa para fazer frente a este grande contigente de alunos.

Nesta situação de características dinâmicas, o planejamento periódico das inúmeras aulas teóricas, laboratório, seminários, palestras etc... que devem ser ministradas semanalmente às turmas de alunos, associadas aos diversos cursos que são oferecidos pela Universidade, se faz necessário.

O problema de planejamento das aulas sob as condições de recursos humanos, físicos e financeiros existentes assume então, um papel relevante para um bom desempenho das atividades de ensino.

O tratamento deste problema, mais precisamente o do planejamento de horários de aulas, utilizando técnicas de Pesquisa Operacional com o auxílio de recursos de computação é colocado como objetivo principal deste trabalho.

Através de uma estratégia concebida neste trabalho foi possível viabilizar a resolução do problema em sua forma ampla que consiste em considerar a programação dos horários como o resultado de uma série de objetivos a serem atingidos, tais como: otimização dos recursos de salas, uniformidade da carga horária dos cursos, minimização dos horários vagos e as preferências dos professores em relação aos dias da semana admitindo o tempo e a frequência das aulas variáveis.

No capítulo I são revistos alguns dos trabalhos desenvolvidos anteriormente e representativos das diferentes estratégias adotadas, em sua maioria realizados na Alemanha e voltados para o problema de programação de horários em escolas secundárias. A otimização dos recursos de salas é também tratada e formulada como um problema de Programação Inteira.

No capítulo II a uniformidade da carga horária e a preferência dos professores quanto aos dias da semana são relacionados com o problema de se alocar as aulas nos dias da semana. Sua resolução é obtida através de um procedimento heurístico.

No capítulo III, a partir da ordenação disciplina-sala-dia de semana obtida nos capítulos I e II, é desenvolvido um procedimento heurístico voltado para determinar uma solução, não necessariamente ótima, para minimização dos horários vagos entre as aulas dos cursos.

No capítulo IV são feitas as conclusões sobre a estratégia adotada e os resultados obtidos e algumas sugestões são propostas para futuros trabalhos.

Finalmente nos apêndices 1 e 2 são apresentados os resultados obtidos da aplicação da estratégia em uma série de experimentos.

CAPÍTULO I

O PLANEJAMENTO DE HORÁRIOS DE AULA

1.1. Apresentação do problema

O planejamento de horários de aulas, corresponde de modo geral, ao problema de ordenação das n-tuplas professor, disciplina, turma nos horários disponíveis das salas de aulas, satisfazendo a certas restrições e atendendo a determinados objetivos.

As restrições fundamentais compreendem:

1. O tempo e a capacidade diária disponíveis das salas de aula.
2. A inexistência de choques de horários de aulas tanto para os professores quanto para as turmas.
3. A existência de um número definido de horas de aulas semanais para cada disciplina.

Enquanto que os principais objetivos são definidos por:

1. As aulas de cada disciplina devem ser distribuidas de maneira uniforme nos dias da semana.
2. A carga horária das turmas deve ser uni

formemente distribuida ao longo da semana.

3. Deve ser minimizado o número de horários vagos para as turmas.

4. Otimizar o uso dos recursos existentes.

Outras restrições tais como: tipo e localização das salas e outros objetivos secundários como a preferência de horários e encadeamento de determinadas disciplinas podem eventualmente ser adicionados mas por conveniência de explanação não serão aqui considerados.

Denominando-se uma combinação a uma ordenação das n-tuplas definidas segundo horários determinados e que satisfaça as restrições acima, pode-se então avaliar e diferenciar as combinações na medida que atendam em maior ou menor escala os objetivos acima identificados.

Embora possa ser considerado um algoritmo de enumeração explícita para avaliar todas as n-tuplas verifica-se que não é possível aplicá-lo por razões práticas e econômicas, desde que o simples caso de uma turma com 10 horas semanais de aula e 1 sala disponível 40 horas/semana determinam 10^{19} possibilidades (A_{40}^{10}).

1.2. Revisão dos trabalhos desenvolvidos

Considerando que este tema já foi abordado anteriormente, torna-se importante resumir as estratégias adotadas e os resultados obtidos. Esta tarefa foi facilitada em parte, pelo trabalho feito anteriormente por Bosler e Frangos[2].

A maioria dos métodos de solução do problema de programação de horários desenvolvidos podem ser classificados de acordo com o esquema da figura 1. Na prática as estratégias de solução se baseiam em pelo menos dois dos métodos indicados.

Métodos de Solução
- heurísticos
  - 1. aleatórios
  - 2. planejados
- algorítmos
  - 3. programação linear
  - 4. teoria dos conjuntos.
  - 5. teoria dos grafos.
  - 6. enumeração.

Figura 1

Métodos de Solução

Nos métodos heurísticos distinguem-se dois procedimentos. Nos métodos classificados em 1 são utilizados processos de troca contínua. Gera-se uma combinação dos elementos por exemplo: disciplina, sala, professor, horário, testando-a em seguida, se violar alguma condição é efetuada uma operação de troca e o procedimento se repete. Alguns trabalhos feitos procuram orientar o processo de troca, de modo mais racional, utilizando-se índices de colisão. Este método foi empregado, segundo Bosler por Lazak[11].

Lazak propõe um processo de troca que se baseia em duas hipóteses:

1. Os preenchimentos dos horários obtidos sem colisões, através da geração aleatória das combinações tem chance, ainda que pequena, de fazerem parte da solução final.

2. Os preenchimentos obtidos através da resolução de colisões tem mais chance de fazerem parte da solução final do que os obtidos sem colisão. Assume-se que cada preenchimento posterior (com alto índice de estágio do processo) é melhor do que um anterior.

As trocas então podem ser feitas considerando a expressão:

$$\text{Mínimo} \sum_{L} \left[KO(L) + IS(L) \right]$$

onde:

KO(L) - índice de colisão do vetor aula L. Indica quantos vetores aulas foram descartados na operação de preenchimento da combinação referente a aula L.

IS(L) - índice do estágio de processamento. Indica quão definitivo é o preenchimento em relação a solução final.

Observa-se que ambos KO e IS indicam de maneira diferente que cada preenchimento posterior é melhor do que o anterior. Por outro lado é difícil garantir, por causa do processo aleatório de geração, que um preenchimento posterior é melhor do que o anterior. Foi questionado também a convergência do método baseado no índice IS(L). Lazak reconhecendo estes fatos propôs um critério de seleção das combinações para obter uma convergencia mais rápida. O resultado é que surge uma contradição entre o processo de seleção, pois este implica em assumir que as combinações definidas inicialmente devem ser as melhores e o processo de trocas que afirma o contrário. Conclui-se também que o aprimoramento desejado provém principalmente do processo de seleção e não do de trocas.

Os métodos em 2 geralmente são conjugados com os métodos de solução algorítmicos. Eles procuram avaliar, através de cálculo prévio, cada combinação de modo que uma vez ela sendo satisfeita não tornem as outras impossíveis de serem satisfeitas. A utilização dos métodos planejados é feita de acordo com a estrutura do problema variando portanto, em função do problema, as etapas que são planejadas e os critérios de resolução adotados.

Nos métodos em 3, os problemas de programação de horários são colocados como um problema de Programação Inteira entretando verifica-se que estas colocações, com raras excessões, tem apenas caráter teórico, sendo as causas da dificuldade de sua aplicação na construção de horários, apontadas como devido à:

- O número de variáveis cresce quadridimensionalmente com os elementos turma x professor x sala x tempo.

- A função objetiva não é univocamente formulável, já que os objetivos secundários não são isentos de contradições.

- As variáveis assumem apenas valores Ø e 1

- O método de branch - and - bound, mesmo para pequenos programas de horários, ~~não~~ ~~for~~nece solução dentro de um tempo aceitável.

Grauss[9] propõe a resolução do problema em três etapas. Na primeira etapa faz-se um plano prévio de preenchimento dos horários. Na segunda é feita a distribuição das aulas nas salas minimizando o número de assentos vazios utilizando-se o modelo de transportes[6]. Finalmente propõe-se a programação de horários através de um método de ordenação.

Em sua formulação, Grauss coloca como restrições do problema as seguintes condições:

a) O tempo disponível das salas de aula

b) O tempo necessário para cada aula

c) A existência de aulas dirigidas por mais de um Instituto (por exemplo: aulas de Fisico-Quimica).

d) A existência de aulas que concentram um número elevado de alunos e as salas individualmente são limitadas em seu tamanho.

e) Não deve haver choques de horários de aulas tanto para professores como para os alunos.

Enquanto que classifica os objetivos como requisitos a serem cumpridos, incluindo:

a) As aulas principais devem ser dadas em dias favoráveis e horários duplos.

b) Não deve haver redução do tempo das aulas importantes.

c) Distribuição uniforme das aulas.

d) Atender os requisitos dos professores quanto às salas apropriadas para suas aulas.

Na primeira etapa então é feito um preenchimento prévio dos horários das disciplinas que conflitem com um esquema geral de programação de horários. No final desta fase, dispõe-se então, de um plano de preenchimento que já divide os intervalos de tempo em determinadas salas de aula.

A etapa seguinte corresponderá a distribuição das aulas restantes nos horários disponíveis das salas, de modo a minimizar os assentos livres. Para isto o problema é formulado segundo o modelo de transportes.

Sendo $\{A_K ; K= \overline{1,p}\}$ o conjunto das aulas

e

$\{S_\ell ; \ell=\overline{1,q}\}$ o conjunto das salas;

o problema pode ser definido através do quadro 1.

Quadro 1

Representação do problema segundo o modelo de Transportes

S_ℓ \ A_κ	A_1		A_p	
S_1	x_{11} c_{11}		x_{1p} c_{1p}	a_1
S_q	x_{q1} c_{q1}		x_{qp} c_{qp}	a_q
	b_1		b_p	

onde:

$x_{\ell\kappa}$: indica o nº de horas da aula κ na sala ℓ

$c_{\ell\kappa}$: nº de assentos vazios se a sala ℓ foi alocada a aula κ

a_ℓ : tempo disponível da sala ℓ por semana

b_κ : horas da aula κ por semana

a formulação fica:

$$\text{Minimizar} \quad K(x) = \sum_{\ell} \sum_{\kappa} x_{\ell\kappa} \, c_{\ell\kappa}$$

$$\text{Sujeito à:} \quad \forall \ell: \quad \sum_{\kappa} x_{\ell\kappa} \leq a_\ell$$

$$\forall \kappa: \sum_{\ell} x_{\ell\kappa} \leqq b_{\kappa}$$

$$\forall \ell,\kappa: x_{\ell\kappa}, c_{\ell\kappa}, a_{\ell}, b_{\kappa} \geq \emptyset$$

Na terceira etapa, a programação dos horários propriamente dita, o autor trata do problema superficialmente. Sugere que se ordene as horas de aulas das disciplinas i na sala ℓ segundo um critério de ordenação simples, admitindo que não deverá haver conflitos de horários por causa dos preenchimentos prévios feitos na etapa 1. Caso haja, poderá ser empregado algum outro processo de ordenação.

Os métodos que utilizam a teoria dos conjuntos procuram a partir das disponibilidades dos elementos turmas e professores, estabelecer condições necessárias e suficientes para a soluçao do problema, representado pela matriz demanda de ensino e pela condição de unicidade, isto é, nenhum professor pode estar simultaneamente em duas salas.

Gotlieb[8] define as condições considerando:

r_{ij}= as aulas da turma c_j com o professor t_i

as condições são:

$$\| \underset{c_j \varepsilon S}{UA^{ij}} \| \geq \sum_{c_j \varepsilon S} r_{ij} \qquad \forall i \text{ e } \forall s \mid S \subseteq C^i \text{ (conjunto das turmas)}$$

$$||\bigcup_{t_j \varepsilon S} A^{ij}|| \geq \sum_{t_j \varepsilon S} r_{ij} \quad \forall j \text{ e } \forall s \mid S \subseteq T^j \text{ (conjunto dos professores)}$$

onde $A^{ij} = A^i \cap A^j$ (interseção das disponibilidades de t_i e c_j).

Verifica-se com simples exemplos que estas condições são necessárias mas não suficientes. Foi feita uma experiência com um exemplo simples de 9 turmas x 9 professores x 9 tempos sob a hipótese de que cada professor atendesse a cada turma pelo menos uma vez. O algorítmo[5] admite um preenchimento prévio e é representado na figura 2. A aplicação à programação reais entretanto é dificultado pelo aumento das pré-ocupações e, pelo tempo de processamento que cresce exponencialmente.

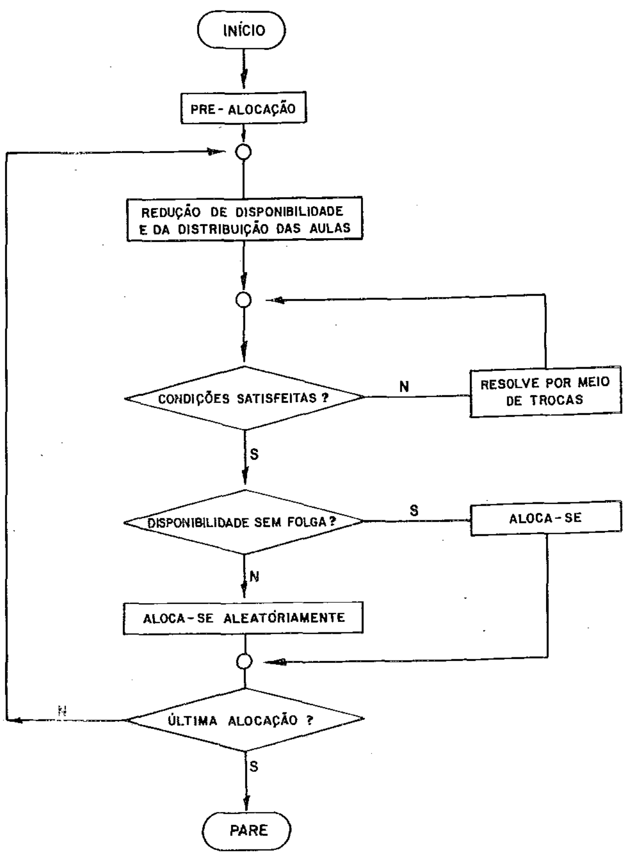


Figura 2

O algorítmo de Csima e Gotlieb

Algumas tentativas foram feitas para se resolver o problema de programação de horários através da teoria dos grafos, em particular os métodos de fluxo em rede[7].

Werra[14] formula o problema nos seguintes termos:

Seja uma escola caracterizada por:

(a) um conjunto de salas $C = \{ c_i \ ; \ i=1, \ldots\ldots, q \}$

(b) um conjunto de professores $M = \{ m_j \ ; \ j=1, \ldots\ldots, p \}$

(c) um conjunto de disciplinas $B = \{ b_\ell \ ; \ \ell=1, \ldots\ldots, s \}$

(d) um conjunto de períodos $N = \{ n_\kappa \ ; \ \kappa=1, \ldots\ldots, n \}$

(e) um conjunto de reuniões U que devem ocorrer durante uma semana.

Cada reunião dura uma hora (um período) e envolve uma ou mais salas, um ou mais professores e uma ou mais disciplinas. Uma programação de horários consiste então de uma alocação de todas as reuniões de U nos períodos disponíveis da semana ou seja, é uma partição de U em subconjuntos $H_1, \ldots, H_n$ onde H_κ é o subconjunto de todas as reuniões que ocorrem no período n_κ (ou simplesmente κ).

O conjunto U é representado por uma matriz de requisitos A (p x q) cujos elementos a_{ij} representam o número de reuniões do professor m_j na sala c_i. Define-se $\alpha_i \leq n$ o número total de reuniões envolvendo a sala c_i e $\beta_j \leq n$ o número total de reuniões envolvendo o professor m_j.

A matriz A pode ser por sua vez representada por um grafo bipartido G com nodos c_i e m_j e cada combinação $c_i - m_j$ por uma aresta interligando c_i e m_j. Os α_i e β_j são definidos como os graus dos vértices.

Supondo que as restrições do problema sejam dadas pelas matrizes E e D definidas por:

$$E = (e_{i\kappa})$$

onde: $e_{i\kappa} = 1$ se a sala i não estiver disponível no período κ

$e_{i\kappa} = \emptyset$ caso contrário.

$$D = (d_{j\kappa})$$

onde: $d_{j\kappa} = 1$ se o professor j não estiver disponível no período κ.

$d_{j\kappa} = \emptyset$ caso contrário

Sejam os graus de liberdades dos elementos c_i, m_j e $c_i - m_j$ definidos por:

$$M_{c_i} = n - \sum_{\kappa} e_{i\kappa} - \alpha_i$$

$$M_{m_j} = n - \sum_{\kappa} d_{j\kappa} - \beta_j$$

$$N_{ij} = \sum_{\kappa} (1-e_{i\kappa})(1-d_{j\kappa}) - a_{ij}$$

A programação dos horários corresponderá então a alocação sucessivas das reuniões nos períodos $\kappa, \kappa+1, \kappa+2, \ldots\ldots, n$ através da resolução de um problema de fluxo em cada período. Para isso a alocação das reuniões H_κ no período κ deve ser tal que o problema residual $\kappa+1, \kappa+2, \ldots.n$, tenha solução. A determinação dos H_κ no período κ pode ser obtida através da resolução em cada período κ de um problema de fluxo máximo com custo mínimo (com certas alterações) em um grafo G_κ. Este é construido inicialmente a partir de G introduzindo uma fonte S ligada aos nodos c_i e um poço T ligado aos nodos m_j e fazendo as capacidades das arestas (S, c_i)e (m_j, T) unitárias e as arestas (c_i, m_j) com capacidades infinitas.

O autor recomenda a utilização da função custo definida por:

$$F(S, c_i) = \begin{cases} \infty \text{ se } c_i \text{ não estiver disponível no período } \kappa. \\ M_{c_i} \quad \text{caso contrário} \end{cases}$$

$$F(m_j, T) = \begin{cases} \infty \text{ se } m_j \text{ não estiver disponível no período } \kappa \\ M_{m_j} \quad \text{caso contrário} \end{cases}$$

$$F(c_i\ m_j) = N_{ij}$$

Nos períodos subsequentes, G'_K é obtido eliminando as combinações alocadas no período anterior.

Werra, observa que na prática é necessário alocar todas as arestas $c_i - m_j$ com $N_{ij} = \emptyset$ (se c_i e m_j estão livres) e depois determinar o fluxo máximo com custo mínimo envolvendo os professores e salas restantes.

O autor verifica que no sistema real é necessário considerar muitos outros requisitos tais como: horário duplo, aulas especiais e aulas distribuidas durante a semana. Sua sugestão é que se faça um preenchimento prévio alocando estes casos particulares. Estas reuniões são então eliminadas do conjunto U e colocando os elementos envolvidos como não disponíveis nos referidos períodos. Foi escrito um programa em Fortran e uma série de esperimentos foram realizados com desempenho considerado satisfatório e cujos resultados são mostrados no quadro 2. As alocações que não foram obtidas são ajustadas manualmente.

Quadro 2

Resultados obtidos por Werra

Salas	Professores	Períodos	Reuniões	Percentagem não alocados	Tempo Processamento (min)
9	15	36	300	0.3	4
34	64	35	1200	3	30
48	84	35	1700	5	50

1.3. Considerações sobre os Trabalhos

Em grande parte os trabalhos feitos foram orientados para o problema de programação de horários das escolas primárias e/ou secundárias que se diferenciam da Universidade em relação à programação de horários nos seguintes aspectos:

(a) As aulas com horários múltiplos (duplos, triplos,etc...) são menos frequentes.

(b) As turmas são praticamente associadas as salas.

(c) Existe maior disponibilidade dos professores e alunos.

(d) Os problemas de distribuição uniforme das cargas horárias e horários vagos se apresentam com menor grau de dificuldade.

Uma crítica que Bosler fez a maioria dos trabalhos é a dificuldade de aplicação dos resultados obtidos na construção de horários reais.

Pode-se inferir que, a formulação do problema de programação de horários, como uma ordenação estrita dos elementos professor, disciplina, sala e tempo e a diminuição da importância dos aspectos acima e de outros com o objetivo de se aplicar técnicas de resolução para se obter solução ótima, resulta que mesmo que seja obtida, a solução não apresentará condições operacionais.

Em se tratando de Universidades, os aspectos acima assumem maior relevância, então por mais razão ainda, não se justifica uma orientação de natureza semelhante. Por outro lado, o problema se apresenta com uma complexidade bem maior. Estas duas razões, justificam o encaminhamento do problema, no sentido de se obter uma solução não ótima mais operacional, para o planejamento de horários de aula em Universidades.

1.4. Estratégia de Solução

Basicamente a estratégia adotada para resolver o problema do planejamento de horários de aulas consiste em decompô-lo em três sub-problemas conforme o esquema da figura 3.

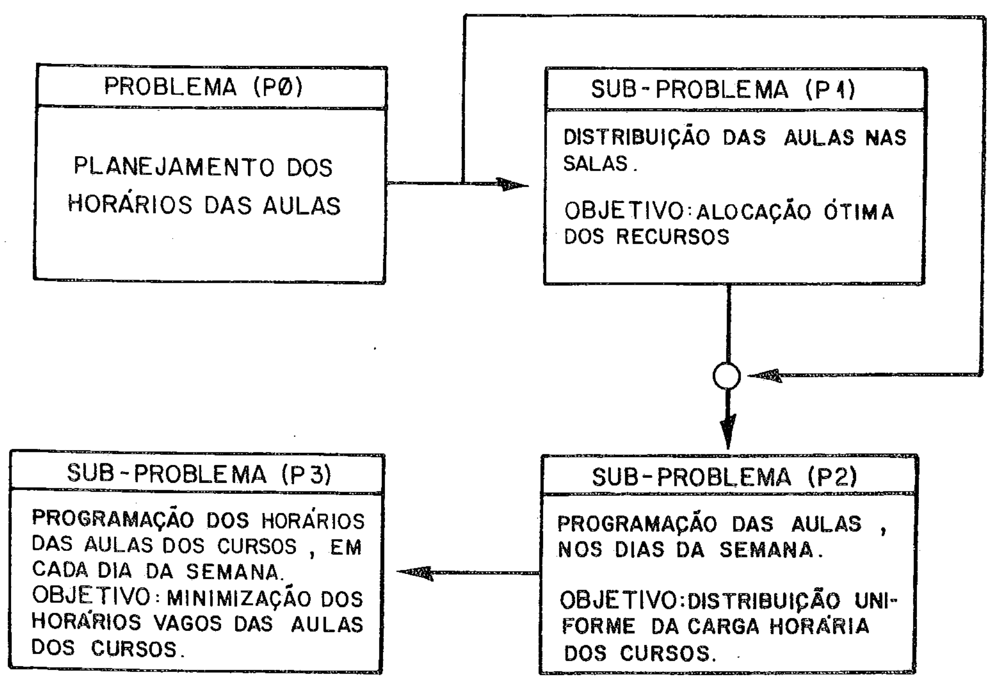


Figura 3
Estratégia de Solução

O sub-problema P_1 corresponde à distribuição das aulas das disciplinas nas salas, de modo a satisfazer o tamanho da turma, o tempo de aula, o tipo de aula e a localização apropriada.

Este problema pode ser formulado como um problema de Programação Inteira, cuja solução tem sido investigada amplamente e existem métodos eficientes para resolvê-la[12] portanto para concentrar os esforços no problema de programação de horários propriamente dito será apenas discutida sua formulação.

Sejam:

t_d: unidades de tempo de cada aula da disciplina d

f_d: a frequência semanal das aulas da disciplina d

v_s: unidades de tempo disponível semanalmente na sala s

x_{ds}: número de aulas alocadas à disciplina d na sala s

tem-se então:

$$\forall\ s: \sum_d t_d\, x_{ds} \leq v_s$$

$$\forall\ d: \sum_s x_{ds} = f_d$$

Os custos c_{ds} de se alocar a disciplina d a sala s podem ser definidos através de um critério que, considerando as disponibilidades dos recursos das salas e os requisitos de capacidade, localização e tipo das disciplinas, determina um peso para cada combinação disciplina-sala.

O requisito de capacidade é absoluto, isto é, não se pode ministrar aulas em salas que não caibam os alunos. O número de assentos livres também podem, dependendo dos recursos, ser penalizados ou não.

O tipo da sala pode ser necessário para determinadas aulas, por exemplo: aulas de laboratórios, mas pode não ser.

Quanto a localização é sempre preferível concentrar as aulas dos cursos próxima aos respectivos departamentos mas não havendo recursos suficientes a localização será distribuida segundo pesos determinados.

Estas considerações sugerem então uma escala de pesos de Ø a um valor M grande a ser definida pelo responsável pelo planejamento acordo com seus recursos e necessidades:

O problema de Programação Inteira resultante é de:

$$\text{Minimizar} \quad F(x) = \sum_{d} \sum_{s} c_{ds}\, x_{ds}$$

sujeito a:

$$\forall s: \sum_{d} t_d \; x_{ds} \leq v_s$$

$$\forall d: \sum_{s} x_{ds} = f_d$$

$$x \geq 0 \text{ inteiro}$$

A ramificação na figura 3 reflete o fato de que em vários casos a distribuição das disciplinas nas salas já se encontra praticamente definida, seja pelo acúmulo das experiências na programação das aulas feitas manualmente ou pela organização existente do espaço físico da Universidade através dos Centros, Institutos e Departamentos.

A partir dos dados de sala, frequência, tempo de aula e preferências de dias das disciplinas, o sub-problema (P2) se resume em distribuí-las ao longo da semana de modo a se obter a uniformidade da carga horária dos cursos, atendendo ao máximo as preferências das disciplinas e satisfazendo as capacidades de tempo diárias das salas.

O sub-problema (P3), de acordo com a programação das aulas nos dias da semana, procederá a programação de horários para cada dia da semana de modo a minimizar os horários vagos entre as aulas dos cursos.

As vantagens que a redução do problema (P4) aos sub-problemas P1, P2 e P3 apresentam são as seguintes:

- Redução do grau de complexidade do problema.
- Permite avaliar melhor o problema.
- Facilita o emprego de técnicas de otimização e/ou heurísticas existentes ou o seu desenvolvimento.
- Identifica-se com maior precisão as fases onde atuam efetivamente os objetivos que se deseja atingir.

C A P Í T U L O I I

PROGRAMAÇÃO DAS AULAS NOS DIAS DE SEMANA

2.1. INTRODUÇÃO

O problema que se coloca ao distribuir as aulas das disciplinas de um curso, caracterizadas pela sala utilizada, frequência semanal-tempo - preferência de dias, nos dias da semana é o de se atingir dois objetivos:

1º Objetivo principal: Obter a uniformidade de carga horária de aulas do curso, durante os dias da semana.

2º Objetivo secundário: Atender a preferência de determinadas disciplinas em relação aos dias da semana.

Sujeito à restrição dada pelo tempo disponível diário das salas de aula utilizadas pelas disciplinas.

2.2. Formulação Matemática

O presente problema envolve a definição dos seguintes conjuntos:

(a) O conjunto de salas: $S = \{ s;\ s = \overline{1,q} \}$

(b) O conjunto de cursos: $C = \{ c;\ c = \overline{1,n} \}$

(c) O conjunto de disciplinas relacionadas com determinado curso c: $D_c = \{ d;\ d = \overline{1,m} \}$

(d) O conjunto dos tempos diários disponíveis das salas: $V = \{ v_s;\ v_s = \overline{1,16} \}$ horas

(e) O conjunto de frequência de aulas: $F = \{ f;\ f = \overline{1,5} \}$

(f) O conjunto de dias da semana: $G = \{ g;\ g = 1$ (SEG), 2 (TER), 3 (QUA), 4 (QUI), 5 (SEX) $\}$

(g) O conjunto de combinações de dias segundo a frequência f: $B = \{ B_f \}$ onde:

$$B_1 = G$$

$$B_2 = \{ (a,b), a < b \quad a,b \in G \}$$

$$B_3 = \{ (a,b,c) \; a < b < c \; a,b,c \in G \}$$

$$B_4 = \{ (a,b,c,d) \; a < b < c < d \\ a,b,c,d \in G \}$$

$$B_5 = \{ (a,b,c,d,e) \; a < b < c < d < e \\ a,b,c,d,e \in G \}$$

(h) Um conjunto de tem pos de aula: $T = \{ t; \; t = \overline{2,8} \}$ (1/2) hora

Sejam então, definidos para cada disciplina $d \subseteq D_c$ a n-tupla (s,f,t) e o subconjunto P_d^* de preferência ordenadas através de:

$$P_d{}^* = \{ P_o \; ; \; P_o \subseteq B_f, \; o = \overline{1, \dim (B_f)} \}$$

onde

o é a posição de P em P_d^*

O problema de programação de aulas nos dias de semana para o curso c correspondente será definido através do seguinte quadro:

Quadro 1

Representação do problema de programação de aulas

Dias g / Salas s	1 SEG	2 TER	3 QUA	4 QUI	5 SEX	tempo diário disponível das salas v_s
1	x_{11} z_{11}				x_{15} z_{15}	V_1
⋮						⋮
q	x_{q1} z_{q1}				x_{q5} z_{q5}	V_q
carga horária →	U_1				U_5	

tem-se que:

$$x_{sg} = \sum_{d \subset D_c} t_{sd} \cdot z_{dg} \leq v_s \quad \forall\, s, g \qquad (1)$$

$$\text{com } z_{dg} = \begin{cases} 1. \text{ Se houver aula da disciplina d no dia g} \\ \emptyset. \text{ Caso contrário} \end{cases}$$

Esta inequação representa o fato de que o preenchimento da sala s no dia da semana g não pode exceder o tempo disponível v_s da sala.

A carga horária do curso c no dia g é dada por:

$$U_g = \sum_{s} x_{sg} \qquad (2)$$

Fazendo

$$z_d = \{ g \in G \mid z_{dg} = 1 \}$$

então a ordem de preferência obtida O_d é:

$$O_d = \text{posição de } z_d \text{ em } P_d^* \qquad (3)$$

A carga horária média

$$\bar{U} = \frac{\sum_{d \in D_c} t_d \times f_d}{5} \qquad (4)$$

Deseja-se então:

$$\text{Minimizar} \left[\begin{array}{l} \Sigma_g (\bar{U} - U_g)^2 / 5 \qquad (5) \\ \qquad e \\ \sum_{d \in D_c} O_d / m \end{array} \right.$$

Sujeito a (1)

2.3. Descrição do algorítmo

A resolução do problema:

$$\text{Minimizar} \left[\begin{array}{l} \sum_{g} (\bar{U} - Ug)^2 / 5 \quad (5) \\ \qquad e \\ \sum_{d \in D_c} O_d / m \end{array} \right.$$

Sujeito à:

$$x_{sg} = \sum_{d \in D_c} t_{sd} \cdot z_{dg} \leq v_s \quad \forall s, g \qquad (1)$$

$$\text{com} \quad z_{dg} = \left[\begin{array}{l} 1. \text{ se houver aula da disciplina } d \text{ no dia } g. \\ \emptyset \text{ caso contrário} \end{array} \right.$$

não é viável através dos métodos convencionais disponíveis. Assim foi concebido um procedimento heurístico descrito a seguir, que obtivesse uma solução não ótima porém operacional para o problema.

Basicamente o procedimento define primeiro, um critério de seleção das disciplinas do curso. Depois para cada disciplina selecionada determina-se o conjunto que define as melhores combinações de dias em relação a uniformidade da carga horária. Faz-se em seguida a interseção deste conjunto com o de preferência das disciplinas e ordena-se o conjunto interseção resultante de modo que se dê prioridade a preferência da disciplina no caso de dois elementos do conjunto contribuirem da mesma forma para a uniformidade da carga horária.

O critério de seleção determina que se dê prioridade às disciplinas de maior frequência de modo a minimizar a ocorrência de violação da restrição do tempo da sala.

Definindo-se o valor de uma combinação L dos dias de G por:

$$W_L = \sum_{g \varepsilon L} Ug \qquad (6)$$

o processo heurístico pode ser resumido da seguinte forma:

PASSO 1: Ordene as disciplinas segundo os maiores valores de frequência.

PASSO 2: Selecione a próxima disciplina segundo a ordem definida no Passo 1. Pare se não houver mais disciplinas.

PASSO 3: Calcule os valores das combinações definidas por B_f de acordo com (6), eliminado as que violaram (1) e ordene-as segundo os menores valores.

PASSO 4: Determine o conjunto interseção entre o conjunto definido no passo 3 e o conjunto de preferências da disciplina.

PASSO 5: Ordene o conjunto interseção segundo os menores valores e segundo a ordem de preferência da disciplina quando dois ou mais elementos tiverem o mesmo valor.

2.4. Aplicação

Seja o conjunto B_f definido pela tabela 1 e as preferências das disciplinas, padronizadas de acordo com a frequência, definidas pela tabela 2.

Tabela 1

Conjunto de Combinações B_f

Freq \ Ord	1	2	3	4	5	6	7	8	9	10
1	5	4	3	2	1					
2	(4,5)	(3,5)	(2,5)	(1,5)	(3,4)	(2,4)	(1,4)	(2,3)	(1,3)	(1,2)
3	(1,3, 5)	(2,4, 5)	(1,4, 5)	(2,3, 5)	(1,2, 5)	(1,3, 4)	(1,2, 4)	(3,4, 5)	(2,3, 4)	(1,2, 3)
4	(1,2, 3,4)	(1,2, 3,5)	(1,2, 4,5)	(1,3, 4,5)	(2,3, 4,5)					
5	(1,2, 3,4, 5)									

Tabela 2

Preferência das disciplinas segundo a frequência de aulas

Ord / Freq	1	2	3	4	5	6	7	8	9	10
1	1	2	3	4	5					
2	(1,3)	(2,4)	(3,5)	(1,4)	(2,5)	(1,5)	(1,2)	(2,3)	(3,4)	(4,5)
3	(1,3, 5)	(1,2, 4)	(1,2, 5)	(1,3, 4)	(1,4, 5)	(2,3, 5)	(2,4, 5)	(1,2, 3)	(2,3, 4)	(3,4, 5)
4	(1,2, 4,5)	(1,2, 3,5)	(1,3, 4,5)	(1,2, 3,4)	(2,3, 4,5)					
5	(1,2, 3,4, 5)									

Considere o curso definido pelas seguintes disciplinas apresentadas no quadro 2 e supõe-se $v_s = 20$ tempos $\forall_s$

Quadro 2
Disciplinas do Curso e seus Parâmetros

DISCIPLINA	SALA	FREQ	TEMPO	PREF
1	1	1	4	TABELA 2
2	1	2	2	
3	1	1	6	
4	1	2	7	
5	2	5	7	
6	1	5	5	
7	2	2	8	
8	3	4	5	
9	3	5	7	

Quadro 3
Disciplinas ordenadas segundo suas frequências

Ordem	Disciplina	SALA	FREQ	TEMPO	PREF
1	5	2	5	7	TABELA 2
2	6	1	5	5	
3	9	3	5	7	
4	8	3	4	5	
5	2	1	2	2	
6	4	1	2	7	
7	7	2	2	8	
8	1	1	1	4	
9	3	1	1	6	

O passo 1 define a ordem de seleção das disciplinas como apresentado no quadro 3.

De acordo com o passo 2 seleciona-se a disciplina 5 e como só existe uma combinação de dias (1,2,3,4,5) a programação das aulas (PA) fica:

Quadro 4
Programação de aulas - 1ª interação

Disciplinas \ Dias	SEG	TER	QUA	QUI	SEX	Preferência O_d	SALA
5	7	7	7	7	7	1	
$U_g \rightarrow$	7	7	7	7	7		

O mesmo acontece para as disciplinas 6 e 9 e a (PA) fica:

Quadro 5

Programação de aulas - 2ª e 3ª interação

Dias / Disciplinas	SEG	TER	QUA	QUI	SEX	O_d	SALA
6	5	5	5	5	5	1	1
5	7	7	7	7	7	1	2
9	7	7	7	7	7	1	3
U_g →	19	19	19	19	19		

A próxima disciplina é a 8. O passo 4 determina o conjunto:

$X_4 = \{(1,2,3,4)\ (1,2,3,5)\ (1,2,4,5)\ (1,3,4,5)\ (2,3,4,5)\}$

com $W_{L=1,5} = 76$. A interseção de X_4 com o conjunto P_8^* de preferências dado pela tabela 2 utilizando FREQ= 4 e de acordo com o critério do passo 4 resulta o conjunto:

$Y_4 = \{(1,2,4,5)\ (1,2,3,5)\ (1,3,4,5)\ (1,2,3,4)\ (2,3,4,5)\}$

O elemento (1,2,4,5) é então selecionado, $O_8 = 1$ e a (PA) resultante é :

Quadro 6

Programação de aulas 4ª interação

Dias / Disciplinas	SEG	TER	QUA	QUI	SEX	O_d	SALA
6	5	5	5	5	5	1	1
5	7	7	7	7	7	1	2
9	7	7	7	7	7	1	3
8	5	5		5	5	1	3
U_g →	24	24	19	24	24		

Selecionando-se a disciplina 2 o conjunto X_5 é dado por:

$$X_5 = \{ (3,5)\ (3,4)\ (2,3)\ (1,3)\ (4,5)\ (2,5)\ (1,5)\ (2,4)\ (1,4)\ (1,2) \}$$

com

$$W_{L=1,4} = 43 \quad e \quad W_{L=5,10} = 48$$

O conjunto Y_5 resultante da interseção de X_5 com P_2^* será:

$$Y_5 = \{(1,3)\ (3,5)\ (2,3)\ (3,4)\ (2,4)\ (1,4)\ (2,5)\ (1,5)\ (1,2)\ (4,5)\}$$

a solução é (1,3), $O_5 = 1$ e a (PA) fica:

Quadro 7

Programação de aulas 5ª interação

Dias / Disciplinas	SEG	TER	QUA	QUI	SEX	O_d	SALA
6	5	5	5	5	5	1	1
2	2		2			1	1
5	7	7	7	7	7	1	2
9	7	7	7	7	7	1	3
8	5	5		5	5	1	3
$U_g \rightarrow$	26	24	21	24	24		

A próxima disciplina é a 4 o conjunto X_6 é

$$X_6 = \{(3,5)\ (3,4)\ (2,3)\ (1,3)\ (4,5)\ (2,5)\ (2,4)\ (1,5)\ (1,4)\ (1,2)\}$$

e

$$W_{L=1,3} = 45,\quad W_{L=4} = 47,\quad W_{L=5,7} = 48 \quad \text{e} \quad W_{L=8,10} = 50 .0$$

conjunto solução Y_6 é:

$$Y_6 = \{3,5)\ (2,3)\ (3,4)\ (1,3)\ (2,4)\ (2,5)\ (4,5)\ (1,4)\ (1,5)\ (1,2)\}$$

determinando a solução (3,5) com $O_4 = 3$ e a (PA) igual a:

Quadro 8

Programação de aulas 6ª interação

Dias / Disciplinas	SEG	TER	QUA	QUI	SEX	O_d	SALA
6	5	5	5	5	5	1	1
2	2		2			1	1
4			7		7	3	1
5	7	7	7	7	7	1	2
9	7	7	7	7	7	1	3
8	5	5		5	5	1	3
U_g →	26	24	28	24	31		

A disciplina 7 é escolhida. O conjunto X_7 é

X_7 = {(2,4) (1,4) (1,2) (3,4) (2,3) (1,3) (4,5) (2,5) (1,5) (3,5)}

e

W = { 48, 50, 50, 52, 52, 54, 55, 55, 57, 59 }

então $Y_7 = X_7$ interseção P_7^* ordenado é:

Y_7 = {(2,4) (1,4) (1,2) (2,3) (3,4) (1,3) (2,5) (4,5) (1,5) (3,5)} e a solução é (2,4) e $0_7 = 2$. A (PA) resulta em:

Quadro 9

Programação de aulas - 7ª interação

Dias / Disciplinas	SEG	TER	QUA	QUI	SEX	0_d	SALA
6	5	5	5	5	5	1	1
2	2		2			1	1
4			7		7	3	1
5	7	7	7	7	7	1	2
7		8		8		2	2
9	7	7	7	7	7	1	3
8	5	5		5	5	1	3
$U_g \rightarrow$	26	32	28	32	31		

A próxima disciplina é a 1. Tem-se que

$$X_8 = \{(1)\ (3)\ (5)\ (4)\ (2)\}$$

$$W = \{26,\ 28,\ 31,\ 32,\ 32\}$$

e $Y_8 \triangleq X_8 \cap P_1^*$ resulta em

$$Y_8 = \{(1)\ (3)\ (5)\ (2)\ (4)\}$$

cuja solução é (1) e $0_1 = 1$

A (PA) fica:

Quadro 10
Programação de aulas - 8ª interação

Dias / Disciplinas	SEG	TER	QUA	QUI	SEX	O_d	SALA
6	5	5	5	5	5	1	1
2	2		2			1	1
4			7		7	3	1
1	4					1	1
5	7	7	7	7	7	1	2
7		8		8		2	2
9	7	7	7	7	7	1	3
8	5	5		5	5	1	3
$U_g \rightarrow$	30	32	28	32	31		

A disciplina 3 é a próxima e última. O conjunto viável X_9 é:

$$X_9 = \{ (3)\ (5)\ (1)\ (2) \}$$

$$W = \{ 28,\ 30,\ 31,\ 32,\ 32 \}$$

e o conjunto solução Y_9 é:

$$Y_9 = \{ (3)\ (5)\ (1)\ (2)\ (4) \}$$

cuja solução é (3) e $O_3 = 3$

Resultando a (PA) em:

Quadro 11

Programação de aulas - 9ª interação

Dias / Disciplinas	SEG	TER	QUA	QUI	SEX	O_d	SALA
6	5	5	5	5	5	1	1
2	2		2			1	1
4			7		7	3	1
1	4					1	1
3			6			3	1
5	7	7	7	7	7	1	2
7		8		8		2	2
9	7	7	7	7	7	1	3
8	5	5		5	5	1	3
$U_g \rightarrow$	30	32	34	32	31		

e

$$U = \frac{\Sigma_g (\bar{U} - U_g)^2}{5} = 1.76 \quad \text{e} \quad O = \frac{\Sigma_d O_d}{9} = 1.5$$

2.5. Testes e Resultados

O algoritmo foi testado com 3 amostras, cada uma representada por 10 cursos. Utilizou-se uma geração aleatória dos parâmetros tempo e frequência das disciplinas. Cada sala ia sendo gerada à medida que nenhuma das salas anteriormente geradas suportassem a carga de tempo (frequência x tempo) de uma disciplina.

Os resultados de cada amostra foram compilados e apresentados no quadro 12. Cada amostra foi testada variando-se a folga do tempo disponível de sala ou seja diminuindo a restrição de violação do tempo das salas.

Observa-se que à medida que se vai relaxando as restrições de tempo de sala obtém-se melhores valores para a função objetiva,como era de esperar. No entanto o mesmo não acontece com as preferências das disciplinas,isto porque estas em determinados passos se contrapõem a uniformidade da carga horária.

É necessário considerar também a influência do processo randômico de geração dos parâmetros das disciplinas. No caso real espera-se um comportamento mais uniforme destes parâmetros o que poderá inclusive sugerir um critério de ordenação inicial das disciplinas mais específico para o problema.

No apêndice 1.1 apenas para uma melhor descrição encontra-se para cada curso da primeira amostra a distribuição das disciplinas nos dias de semana.

No apêndice 1.2 são apresentadas as matrizes curso x sala por dia da semana. Os elementos das matrizes indicam o tempo que cada curso requer em cada sala. Estas matrizes serão os dados de entrada para a programação de horários que será discutido no capítulo seguinte.

No apêndice 1.3 são descritos os parâmetros de chamada das rotinas utilizadas.

Quadro 12

Aplicação do algorītmo de Programação de aulas nos dias da semana ā 3 amostras variando-se a folga do tempo das salas.

AMOSTRA 1		TEMPO SEMANAL DAS SALAS=80 TEMPO DIÁRIO DISPONÍVEL=16 FOLGA=Ø			TEMPO SEMANAL DAS SALAS=80 TEMPO DIÁRIO DISPONÍVEL=20 FOLGA=4			TEMPO SEMANAL DAS SALAS=80 TEMPO DIÁRIO=24 FOLGA=8		
CURSO	DISCIPLINAS (m)	CARGA HORÁRIA MÉDIA ($\bar{U}$)	VARIANCIA (U)	MÉDIA DAS PREFERÊNCIAS (O)	CARGA HORÁRIA MÉDIA ($\bar{U}$)	VARIANCIA (U)	MÉDIA DAS PREFERÊNCIAS (O)	CARGA HORÁRIA MÉDIA ($\bar{U}$)	VARIANCIA (U)	MÉDIA DAS PREFERÊNCIAS (O)
1	9	31.8	1.76	1.56	31.8	6.56	1.33	31.8	1.76	1.56
2	7	15.6	5.44	3.43	15.6	5.44	3.43	15.6	5.44	3.43
3	7	17.2	1.76	3.29	17.2	1.76	3.29	17.2	1.76	3.29
4	7	25.6	6.24	1.86	25.6	1.44	1.57	25.6	1.44	1.57
5	6	14.6	5.04	1.83	14.6	5.04	1.83	14.6	5.04	1.83
6	6	16.8	5.76	2.5	16.8	4.56	2.83	16.8	4.56	4.00
7	7	15.8	13.36	1.86	15.8	2.16	2.43	15.8	2.16	2.14
8	7	15.6	9.04	3.14	15.6	1.84	3.86	15.6	1.04	4.14
9	7	16.6	34.24	1.43	16.6	11.84	1.57	16.6	3.44	2.14
10	9	17.0	24.4	3.56	17.0	1.20	3.00	17.0	1.2	3.00

AMOSTRA 2		TEMPO SEMANAL DAS AULAS=80 TEMPO DIÁRIO DISPONÍVEL=16 FOLGA=Ø			TEMPO SEMANAL=80 TEMPO DIÁRIO=20 FOLGA=4			TEMPO SEMANAL=80 TEMPO DIÁRIO=24 FOLGA=8		
CURSO	DISCIPLINAS (m)	CARGA HORÁRIA MÉDIA ($\bar{U}$)	VARIANCIA (U)	MÉDIA DAS PREFERÊNCIAS (O)	CARGA HORÁRIA MÉDIA ($\bar{U}$)	VARIANCIA (U)	MÉDIA DAS PREFERÊNCIAS (O)	CARGA HORÁRIA MÉDIA ($\bar{U}$)	VARIANCIA (U)	MÉDIA DAS PREFERÊNCIAS (O)
1	8	19.8	12.56	3.63	19.8	6.96	4.38	19.8	6.96	4.38
2	9	21.4	5.04	4.22	21.4	3.84	3.44	21.4	3.84	3.44
3	8	30.2	8.56	3.25	30.2	5.76	3.75	30.2	8.56	3.25
4	8	16.6	5.84	3.88	16.6	2.64	3.25	16.6	2.64	3.25
5	6	18.2	4.16	3.17	18.2	0.56	3.0	18.2	0.96	4.00
6	6	23.4	12.64	1.5	23.4	12.64	1.5	23.4	4.64	2.00
7	7	23.4	3.44	2.57	23.4	3.44	2.57	23.4	3.44	2.57
8	9	27.2	3.76	2.67	27.2	5.76	4.11	27.2	3.76	2.67
9	6	24.6	7.04	2.67	24.6	7.04	2.67	24.6	7.04	2.5
10	8	19.2	8.56	1.88	19.2	4.96	2.25	19.2	0.16	2.13

AMOSTRA 3		TEMPO SEMANAL DAS AULAS = 80 TEMPO DIÁRIO DISPONÍVEL= 16 FOLGA=Ø			TEMPO SEMANAL=80 TEMPO DIÁRIO=20 FOLGA=Ø			TEMPO SEMANAL=80 TEMPO DIÁRIO=24 FOLGA=8		
CURSO	DISCIPLINAS (m)	CARGA HORÁRIA MÉDIA ($\bar{U}$)	VARIANCIA (U)	MÉDIA DAS PREFERÊNCIAS (O)	CARGA HORÁRIA MÉDIA ($\bar{U}$)	VARIANCIA (U)	MÉDIA DAS PREFERÊNCIAS (O)	CARGA HORÁRIA MÉDIA ($\bar{U}$)	VARIANCIA (U)	MÉDIA DAS PREFERÊNCIAS (O)
1	6	17.2	10.16	2.17	17.2	3.76	1.5	17.2	3.76	1.50
2	8	25.8	11.76	1.88	25.8	7.76	2.25	25.8	7.76	1.5
3	7	11.6	7.44	3.0	11.6	7.44	3.43	11.6	2.64	3.71
4	6	18.2	4.96	2.0	18.2	4.96	2.Ø	18.2	4.96	2.0
5	8	25.8	1.76	3.63	25.8	1.76	3.63	25.8	1.76	3.63
6	8	24.0	5.2	3.13	24.0	8.4	3.0	24.0	8.40	3.00
7	9	27.0	7.6	2.67	27.0	7.6	2.67	27.0	7.60	2.67
8	7	23.8	3.76	2.86	23.8	4.96	1.57	23.8	4.96	1.57
9	8	22.0	0.8	3.38	22.0	0.8	2.75	22.0	0.80	3.25
10	7	8.0	2.4	2.71	8.0	3.2	2.86	8.0	0.80	2.14

2.6. Resumo do Programa

O programa para a programação de aulas nos dias da semana foi desenvolvido utilizando a linguagem ALGOL do Burroughs 6700.

O programa está subdividido em 5 rotinas: Principal, Alternativas, Ordena, Solução e Gera, cujas funções e diagrama de blocos são resumidos a seguir.

2.6.1. Rotina: Principal

Esta rotina é a que coordena as chamadas das outras rotinas. O diagrama a seguir define sua organização. Observa-se que após o último curso é impresso a matriz curso x sala por dia da semana que servirá como dados de entrada para o algorítmo de horários descrito no capítulo III.

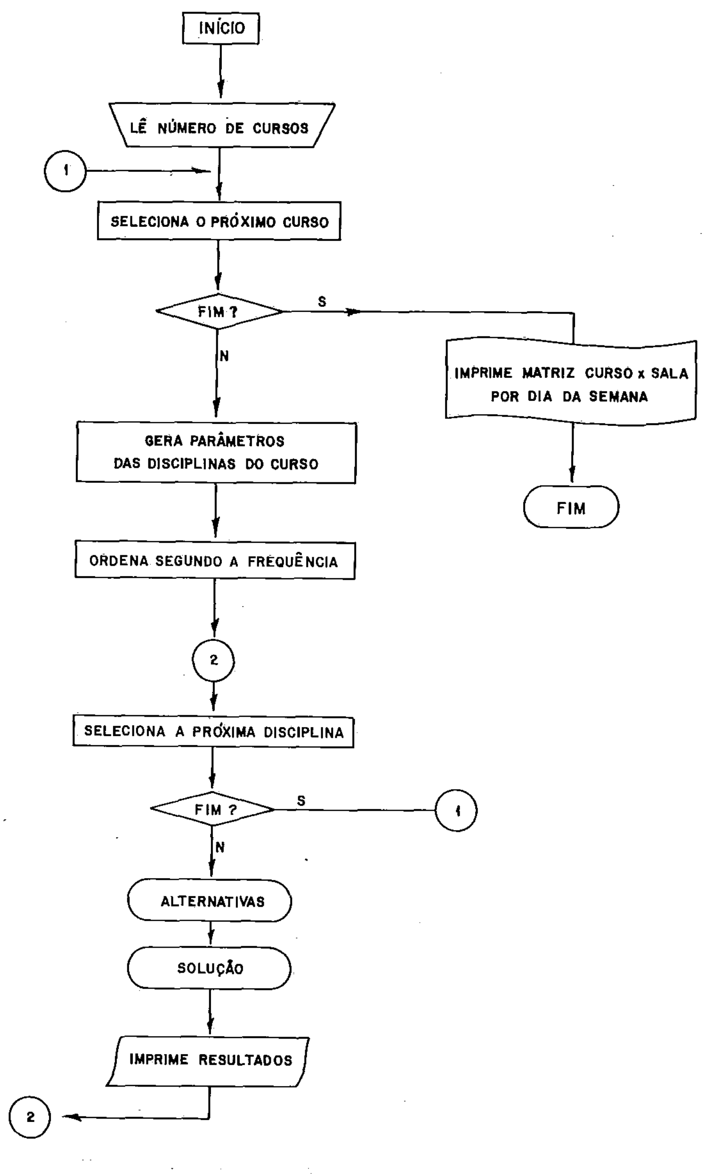


Figura 1
Diagrama da Rotina - Principal

2.6.2. Rotina: Alternativas

Calcula as alternativas viáveis para uma dada frequência segundo a carga horária, o conjunto de preferência e a capacidade diária das salas fornecidas. Se não houver seleciona as que violam menos.

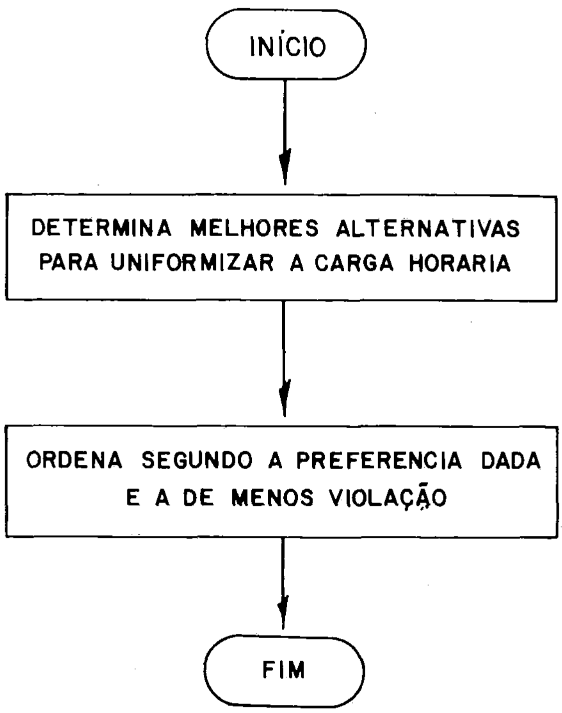


Figura 2
Diagrama da Rotina - Alternativas

2.6.3. Solução

Calcula a interseção de dois conjuntos ordenados, um considerado principal (CP) e o outro secundário (CS).

A interseção é feita segundo a ordem dada pelo conjunto principal ou seja, verifica-se se existe o primeiro elemento do CP em CS, o segundo, etc ... quando existir, o elemento é rotulado com a ordem dada de CS. O conjunto resultante é o con é o conjunto interseção (CI).

O conjunto solução é gerado a partir de CI através da reordenação de seus elementos. A troca de posição de dois elementos é feita quando ambos tem associado o mesmo valor de tempo e o elemento posterior tem ordem inferior (rótulo menor) ao elemento anterior.

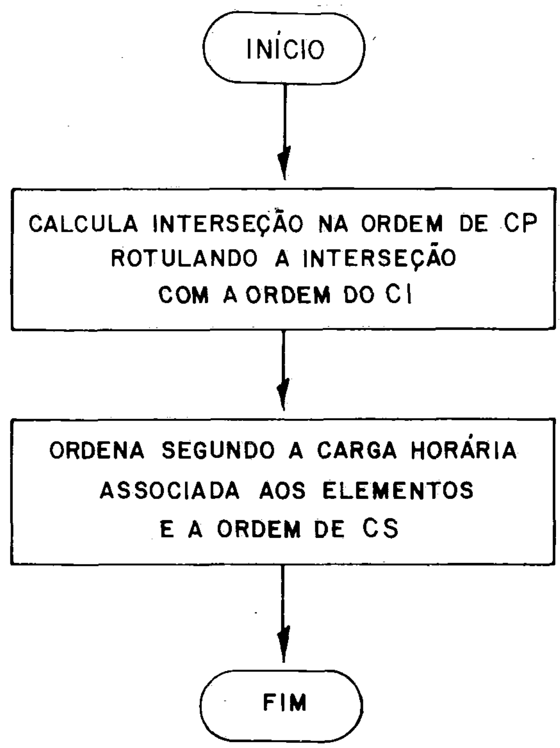


Figura 3
Diagrama da Rotina - Solução

2.6.4. Rotina: Gera

Gera aleatoriamente os parâmetros de um conjunto de disciplinas correspondentes a um curso.

Os parâmetros compreendem:

frequência f: $5 \leq f \leq 1$

tempo t: $8 \leq t \leq 2$

O número da sala é inicializado na rotina principal e é gerada sequencialmente à medida que nenhuma das salas anteriormente geradas suportem a quantidade de tempo da disciplina. O tempo disponível das salas é assumido 80 tempos de 1/2 hora e o número de disciplinas de um curso é gerado aleatoriamente e está compreendido entre 4 e 14.

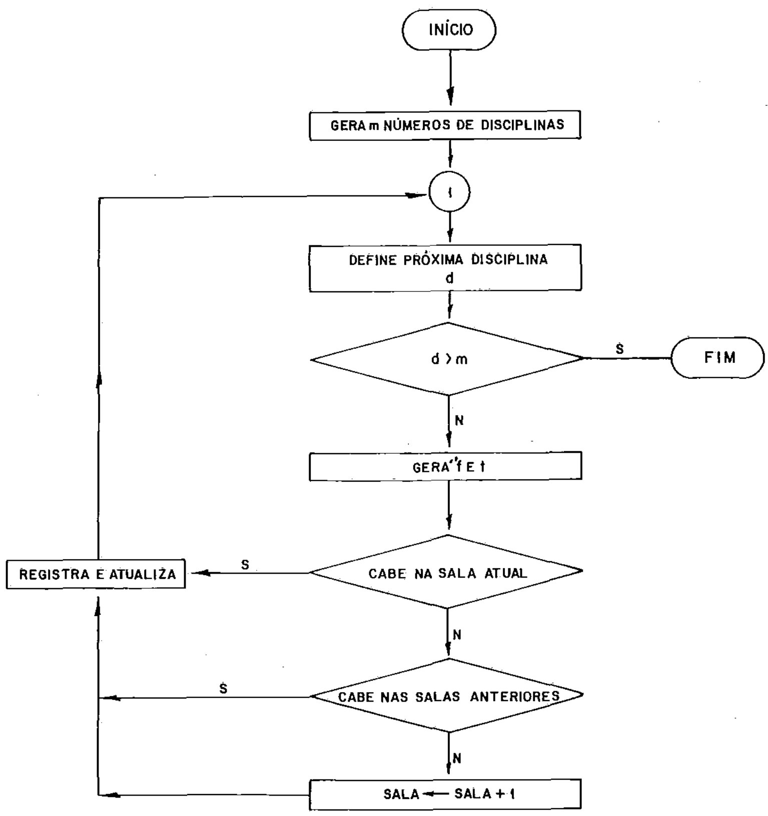


Figura 4
Diagrama da Rotina - Gera

2.6.5. Rotina: Ordena.

Esta rotina ordena um vetor em ordem crescente ou decrescente de valores de acordo com a opção selecionada.

C A P Í T U L O I I I

PROGRAMAÇÃO DOS HORÁRIOS DAS AULAS (PHA)

3.1. INTRODUÇÃO

Obtida a distribuição das aulas nos dias da semana surge, segundo a estratégia adotada, o problema de se definir os horários das aulas.

Através da matriz de tempos de aula por curso, sala segundo o dia da semana como ilustrada no quadro 6, pode-se descrever o referido problema como se segue.

Quadro 1

Matriz tempo de aula por curso, sala e dia de semana

Sala / Curso	1	....	q
1	t_{11}		t_{1q}
⋮			
n	t_{n1}		t_{nq}

Neste quadro o tempo de aula $t_{c,s}$ do curso $\underline{c}$ na sala $\underline{s}$ representa eventualmente o tempo de aula de mais de uma disciplina.

No caso geral tem-se que as salas são compartilhadas entre vários cursos e estes tem aulas em mais de uma sala. Neste caso a programação de horários envolve a resolução das possíveis colisões de horários entre dois ou mais cursos. A colisão ocorrerá quando um curso não conseguir iniciar sua aula em determinada sala porque ela está sendo utilizada por aulas de outro curso. É muito provável a ocorrência de colisões se forem definidas sequências aleatórias de ocupação das salas, além de que o número de possibilidades é extremamente grande, basta considerar que um curso com aulas em $\underline{q}$ salas este número será dado pela permutação de q, supondo $\underline{n}$ cursos ter-se-ia $\prod_{c=1,n} q_c!$

Se houver uma colisão de horários entre dois cursos um deles terá que esperar o outro terminar suas aulas. A soma dos tempos que um curso deve esperar como consequência de sua ocupação das salas é definido como seu tempo de espera. O tempo vago de uma programação de horários para um conjunto de cursos é definido como a soma de todos os tempos de espera dela resultante. O problema de programação de horários corresponde portanto a determinação, para cada curso, de uma sequência de ocupação das salas de modo a minimizar o tempo vago final.

3.2. Formulação Matemática

Dado $C = \{ c; c = \overline{1,n} \}$

$S = \{s;\ s = \overline{1,q}\}$

$r_{c,s}$ = tempo de aula do curso c na sala s

t_s = tempo disponível diário da sala s

Seja:

$S_c = \{s \mid r_{c,s} > \emptyset\}$, o conjunto das salas alocadas ao curso c.

$|S_c|$ = nº de elementos de S_c

$Q_\kappa^c = \{p_1, p_2, p_3 \ldots p_{|S_c|}\}$ a κ-ésima permutação das $|S_c|$ salas de S_c e

$Q^c = \{Q_\kappa^c;\ \kappa = 1,m\}$ onde m = nº de permutações de $|S_c|$ salas, é o conjunto das sequências das salas alocadas ao curso c, obtido pelas permutações das salas.

Define-se para cada sequência Q_κ^c:

(a) O intervalo de ocupação do curso c na sala s

$$I_{c,s}^\kappa = [a_{c,s}^\kappa \quad b_{c,s}^\kappa] \text{ onde}$$

$a_{c,s}^\kappa$ = hora inicial de ocupação da sala <u>s</u> pelo curso <u>c</u>

$b_{c,s}^\kappa$ = hora final de ocupaçaõ da sala <u>s</u> pelo curso <u>c</u>.

$$b_{c,s}^\kappa = a_{c,s}^\kappa + r_{c,s}^\kappa$$

b) O tempo vago da sequência Q_K^c

$$E_c^K = \sum_{J=1,|S_c|-1} e_{c,j}^K$$

sendo

$e_{c,j}^K = a_{c,j+1}^K - b_{c,j}^K$, o tempo vago entre dois elementos sucessivos J e J+1 da sequência Q_K^c.

O problema da programação de horários vagos pode ser então formulado:

Minimizar $\sum_c \sum_j (a_{c,j+1}^K - b_{c,j}^K) z_{c,K}$ sujeito a:

(interseção de aulas) (1) $\forall$ c,K: $(a_{c,s}^K + r_{c,s}^K)\, z_{c,K} < a_{c',s}^{K'} \cdot z_{c',K}$ ou

$(a_{c's}^{K'} + r_{c',s}^{K'})\, z_{c',K} < a_{c,s}^K \cdot z_{c,K}\ \forall_{c'} \neq c$

(disponibilidade) (2) $\forall$ c,K: $(a_{c,|S_c|}^K + r_{c,|S_c|}^K)\, z_{c,K} \leq t_s$

(configuração) (3) $\forall$ c,K : $\sum_{K=1,m_n} z_{c,K} = 1$

$\forall c,K\ z_{c,K}$ booleana e $\forall_c, \forall_s: a_{c,s} \geq \emptyset$

3.3. Descrição do algorítmo

O problema apresentado em 3.2 pode ser colocado na forma mais genérica substituindo cursos por tarefas e salas por indivíduos.

A dificuldade de se resolver este problema tem sido reconhecida[7] desconhecendo-se na literatura atualmente disponível um algorítmo para resolvê-lo. Além disso as particularidades do problema de horários de aula concorreram para que se optasse pelo desenvolvimento de um procedimento heurístico que não necessariamente obtivesse o ótimo mas que permitisse operacionalizar de maneira eficiente a programação de horários em vez de resolver o problema geral.

A principal característica é que a matriz de cursos x sala é esparsa, podendo-se assumir que:

a) não mais do que 5 cursos compartilham de uma mesma sala.

b) as aulas de um curso são distribuidas no máximo em 4 salas diariamente.

Outra característica importante e que deve ser explorada é de que a matriz curso x sala pode eventualmente ser subdivida em matrizes menores ou partições. Como por exemplo considere a figura a seguir.

MATRIZ CURSO x SALA

CURSO \ SALA	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	14	7	7													
2		4		11												
3					7	8										
4						2	8	6	8							
5							4	5	2							
6										6	9					
7											4	8				
8												2	11	3		
9				3										7	3	
10										4					10	6

Figura 1

Exemplo de matriz curso x tempo

Pode-se definir duas paritções P_1 e P_2 onde

$P_1 = [\ t_{cs}\ ];\ c = \overline{1,2,6,10}$ e $s = \overline{1,4,10,16}$ e

$P_2 = [\ t_{cs}\ ];\ c = \overline{3,5}$ e $s = \overline{5,9}$

Existiria ainda a possibilidade de se obter de P_1 uma partição P_3 envolvendo os cursos 1 e 2 se pudesse ser obtido a transferência da aula do curso 9 na sala 4 para outra sala entre à 10 e a 16. Outra hipótese razoável é que o número de cursos de uma partição não ultrapasse a 10.

Emboras estas hipóteses não sejam utilizadas como restrições para o algorítmo descrito a seguir, verifica-se que sob estas circunstâncias ele apresente resultados satisfatórios.

O algoritmo obedece a cinco etapas:

[1ª] Determina-se uma partição da matriz curso x sala dada.

[2ª] Define-se uma ocupação inicial das salas para a partição selecionada.

[3ª] A partir de uma ocupação inicial o algoritmo encontra uma solução que minimize o tempo vago.

[4ª] Procede-se a uma pós-otimização caso o tempo vago seja maior que zero.

[5ª] Se houver tempo vago retorna-se a etapa 2 caso contrário vai-se para [1ª] etapa.

Na primeira etapa as partições são identificadas para serem utilizadas nas etapas seguintes.

A segunda etapa é onde se realiza o procedimento heurístico propriamente dito. Para cada sala que é compartilhada seleciona-se o curso que irá ocupá-la inicialmente. A justificativa deste procedimento reside nas seguintes constatações:

a) As colisões devem ser evitados antecipadamente.

b) O problema torna-se operacional, dentro dos limites estabelecidos, à medida que se puder reduzir o número de possibilidades $\prod_{c=1,n} q_c$ para $\prod_{c=1,n} (q_c-1)$, assim o caso de

10 cursos distribuidos cada um em 3 salas reduz-se de 6^{10} possibilidades para 2^{10} ou seja de um fator de 3^{10}!

O critério de seleção determina que entre dois ou mais cursos que compartilham uma sala terá prioridade aquele que enquanto estiver ocupando a sala, os outros tenham outras alternativas de sala. Se houver mais de um curso escolhe-se o que oferece: 1º o maior número de alternativas

2º alternativas de menor intervalo de tempo.

Quando envolverem mais de dois cursos e os critérios 1º e 2º não discriminarem toma-se a variância dos números de alternativas e dos intervalos. Se persistir o empate escolhe-se o de menor tempo de ocupação e finalmente o primeiro deles. As salas que não são compartilhadas são imediatamente associadas ao único curso pretendente.

Considere para efeito de maior clareza na exposição a partição 1 da matriz curso x sala anterior.

CURSO \ SALA	1	2	3	4	10	11	12	13	14	15	16
1	14	7	7								
2		4		11							
6					6	9					
7						4	8				
8							2	11	3		
9				3					7	3	
10					4					10	6

Figura 2

Partição 1 da matriz

onde as salas compartilhadas são: 2,4,10,11,12,14 e 15, então aplicando-se o critério em cada uma destas salas resultará:

1) Sala 2: Compartilhada pelos cursos 2 e 1.

Verifica-se que se for alocada ao curso 1 oferece 1 alternativa para o curso 2, isto é, enqüanto estiver a sala 2 ocupada o curso 2 utilizará a sala 4.

Supondo agora a sala 2 alocada ao curso 2 verifica-se que o curso 1 terá duas alternativas, a sala 1 ou a sala 3. O 1º critério determina que se escolha o curso 2. Para indicar a escolha fez-se um círculo no elemento $t_{2,2}$ da figura 2.

2) Sala 4: cursos 2 e 9.

Alocando a sala 4 ao curso 2, a alternativa do curso 9 será as salas 14 e 15 perfazendo 10 tempos que não serão suficientes, visto que a sala 4 ao término deste período ainda estará sendo ocupada pelo curso 2, obrigando-o a esperar 1 tempo.

Se for alocada ao curso 9, o curso 2 terá a alternativa da sala 2 sem nenhum tempo de espera. Assim, de acordo com o primeiro critério o curso 9 terá prioridade.

O critério 1 aplicado às salas 10,11,14 e 15 será suficiente para se obter as prioridades desejadas.

(3) Sala 12: cursos 7 e 8

Neste caso o primeiro critério não é capaz de discriminar o curso prioritário já que cada um oferece uma alternativa ao outro. Pode-se então aplicar o segundo critério.Os intervalos de tempos definidos pelas alternativas são calculados do seguinte modo:

alternativa 1: curso 7 na sala 12: 8 tempos e o curso 8 na sala 13: 11 tempos.

intervalo = 11-8 = 3 tempos

alternativa 2: curso 8 na sala 12: 2 tempos e o curso 7 na sala 11: 4 tempos.

intervalo = 4-2 = 2 tempos

resultando de acordo com o critério 2, na escolha da alternativa 2, isto é, selecionando-se o curso 8 para a sala 12.

Na terceira etapa do algorítmo procede-se através de uma técnica de enumeração explícita a partir de uma ocupação inicial das salas.

Seja então a ocupação inicial das salas da figura 2, utilizando-se a notação curso (sala) dada por:

ocupação inicial: 1(1), 2(2), 6(10), 7(11), 8 (12), 9 (4) e 10 (16)

A enumeração é feita através do procedimento descrito a seguir.

Inicialmente seleciona-se um curso que tenha tido preferência de sala e escolhe-se sua primeira sequência por exemplo, o curso 2 é selecionado e a sequência de salas (2→4) escolhida. O símbolo (→) identifica a sequência de ocupação das salas. Em seguida são identificados os cursos que compartilham salas com 2, no caso 1 e 9. Para estes cursos são escolhidas as melhores sequências respeitando a sequência já definida para o curso 1. Se o tempo vago resultante das 2 melhores sequências para 1 e 9 for maior que zero, o algorítmo selecionará outra sequência, se houver para o curso 2 e escolherá outras melhores se quências para 1 e 9 e assim por diante obtendo-se no final o menor tempo vago.

Uma sequência de salas para um curso é melhor do que uma outra se o tempo vago de uma é menor de que o da outra. Caso contrário ambas são iguais e para efeito de escolha do algorítmo seleciona-se a primeira.

Assim sendo as sequências (1→2→3) e (4 →14→15) são selecionadas para os cursos 1 e 9 respectivamente e o tempo vago resultante igual a zero.

O curso 2 é então considerado rotulado e o algorítmo irá em seguida rotular os cursos 1 e 9.

O estado atual de resolução do problema pode ser representado por meio de uma árvore, onde os nodos são os cursos e os ramos definem a existência do compartilhamento de sala. Utilizando-se ainda a notação κ[e] adjacente ao nodo para indicar a sequência κ do curso escolhida e o tempo de espera obtido. A programação de horários resultante é esquematizado pelo quadro de ocupação das salas correspondente. Então a figura 3 representa o estado atual de resolução deste problema.

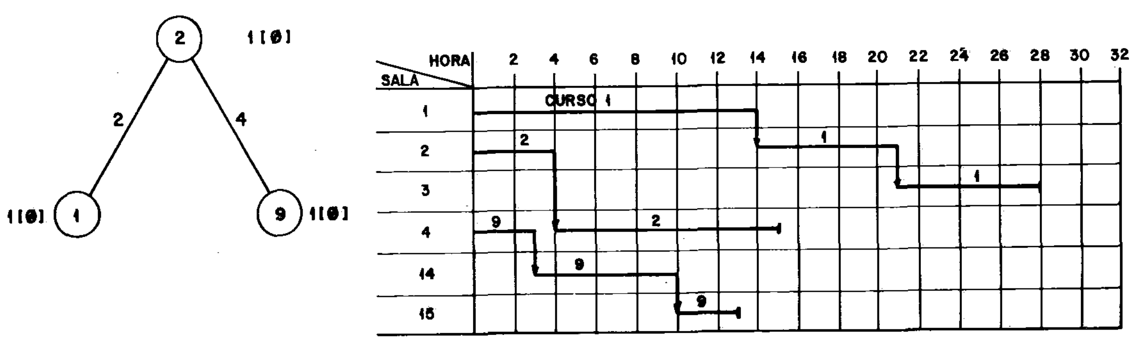


Figura 3
Programação de Horários - 1ª interação

O Curso 1 não compartilha salas com nenhum outro curso além do 2 que já foi rotulado. Rotula-se então o curso 1.

O Curso 9 compartilha as salas 14 e 15 com os cursos 8 e 10 respectivamente.

É escolhida a sequência 1 (12→13→14) para o curso 8.

O curso 10 tem duas possibilidades:

1ª (16→10→15) com e = 3, devido a colisão com o curso 9 na sala 15;

2ª (16→15→10) com e = 7 pelo mesmo motivo.

O algorítmo então volta ao curso 9 e seleciona a sequência 2 (4→15→14) com e = ∅. A sequência 1 do curso 8 é mantida e é escolhida a sequência 2 para o curso 10 com e = ∅ desta vez. O curso 9 é rotulado. A figura 4 representa o estado atual do problema.

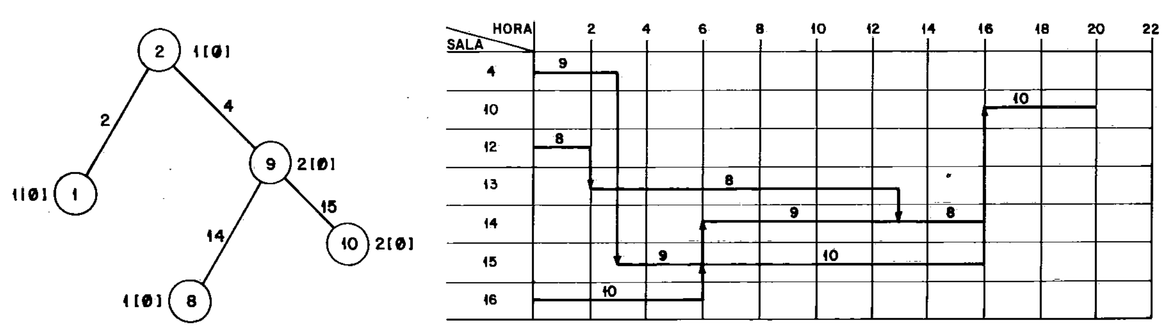


Figura 4
Programação de Horários após a 3ª interação

O curso 8 compartilha a sala 12 com o curso 7. A sequência 7 (11→12) com e=∅ é escolhida para 7. O curso 8 é rotulado.

O curso 10 compartilha a sala 10 com o curso 6. Escolhe-se a sequência 1(10→11) com e=∅ para o curso 6.

O algorítmo prosseguindo irá rotular 7 e 6. O resultado é mostrado na figura 5 correspondendo a programação de horários final com o tempo vago obtido igual a zero. O número de sequências para a etapa seguinte foi reduzida de 10368 para 128.

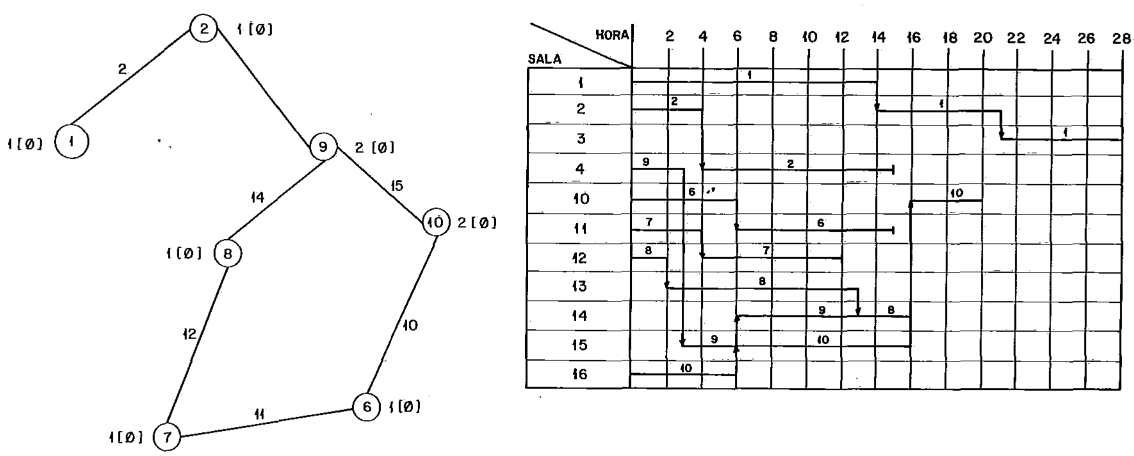


Figura 5

Programação de Horários - 7ª interação

A quarta etapa do algorītmo ē aplicada quando o tempo vago resultante ē maior que zero. Pressupondo-se que a hora inicial das aulas não precisa ser a mesma para todos os cursos,pode-se então verificar para os cursos que tenham tempo de espera, se ē possīvel evitā-lo atravēs de uma alteração na hora inicial de suas aulas. Por exemplo, a programação de horārios para a matriz da figura 6.A ē mostrada na figura 6.B. Observa-se que o tempo de espera do curso 7 ē igual a 2, no entanto se for alterado o inīcio das aulas de Ø hora para 2 horas,obtēm-se o tempo vago igual a zero.

CURSO \ SALA	11	12	13	14	15	16
7	6	9				
8		2	6	3		
9				7	10	
10		5				8

Figura 6 A

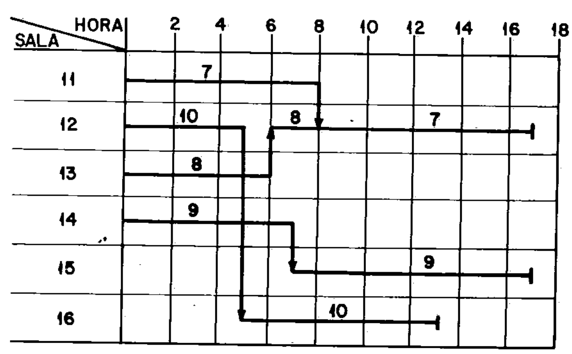


Figura 6 B

Pōs otimização da Programação de horārios

3.4. Testes e Resultados

O algorítmo foi aplicado às matrizes curso x sala referidas no capítulo II.5, obtendo-se as respectivas programações de horários com o tempo vago igual a zero. Com o propósito de se comentar alguns aspectos decorrentes da aplicação do algorítmo, a matriz curso x sala da segunda feira da amostra 1 (apêndice 2) é aqui reproduzida.

SEGUNDA-FEIRA

CURSO \ SALA	1	2	3	4	5	6	7	8	9	10	11	12
1	13	17										
2			15									
3				14	3							
4					13	14						
5					4	3	7					
6							6	15				
7								6	13			
8									5	6		
9										10	15	
10											5	10

PROGRAMAÇÃO DE HORÁRIOS

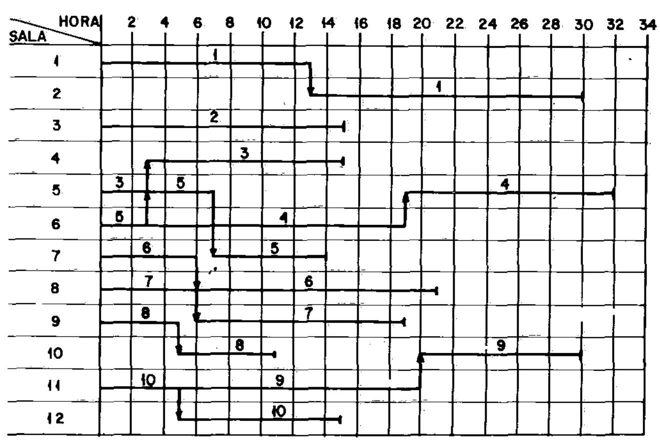


Figura 7
Matriz curso x sala da segunda feira da amostra 1

a) Na primeira etapa verifica-se que os cursos 3,4 e 5 compartilham a sala 5. A escolha do curso prioritário foi feita considerando as seguintes hipóteses:

1º Alocando-se a sala 5 ao curso 3. Obtém-se:

curso 4; 1 alternativa (sala 6, e = ∅)

curso 5: 2 alternativas(salas 6 e 7, e = ∅)

2º Alocando-se a sala 5 ao curso 4. Obtém-se

curso 3: 1 alternativa (sala 4, e = ∅)

curso 5: 1 alternativa(salas 6 e 7, e = 3)

3º Alocando-se a sala 5 ao curso 5. Obtém-se

curso 3: 1 alternativa (sala 4, e = ∅)

curso 4: 1 alternativa (sala 6, e = ∅)

Então pelo critério 1 o curso 3 tem prioridade, pois oferece o maior número de alternativas = 3.

b) Na etapa 5 do algorítmo as horas iniciais dos cursos 4 e 9 foram alteradas de 3 a 5 tempos respectivamente possibilitado obter o tempo vago igual a zero.

Observa-se na maioria dos casos que os tempos vagos iguais a zero foram obtidos devido ao remanejamento das horas iniciais dos cursos.

Embora nem sempre possa utilizar este artifício plenamente devido à restrição do tempo disponível das salas.

c) As violações dos tempos disponíveis das salas 2,6 e 11 pelos cursos 1, 4 e 9, são explicadas a partir das cargas horárias médias iguais a 31,8, 25,6 e 16,6 e das variâncias 1,76 6,24 e 34,24 dos referidos cursos, devido provavelmente ao processo de geração dos parâmetros das disciplinas como observado anteriormente no capítulo 2.5.

3.5. Implementação do Algorĩtmo

Analisando o algorĩtmo verifica-se que ele determina uma rotina principal e quatro rotinas bãsicas que podem ser denominadas: PRINCIPAL, PARTIÇÃO, OCUPAÇÃO, TEMPO VAGO e REMANEJAMENTO cujas funções e diagramas de bloco serão apresentados a seguir.

3.5.1. Rotina: PRINCIPAL

Sua função ẽ coordenar as chamadas das outras rotinas para obter a programação de horãrios a partir da matriz curso x sala lida. O diagrama de blocos de rotina ẽ mostrado na figura 8.

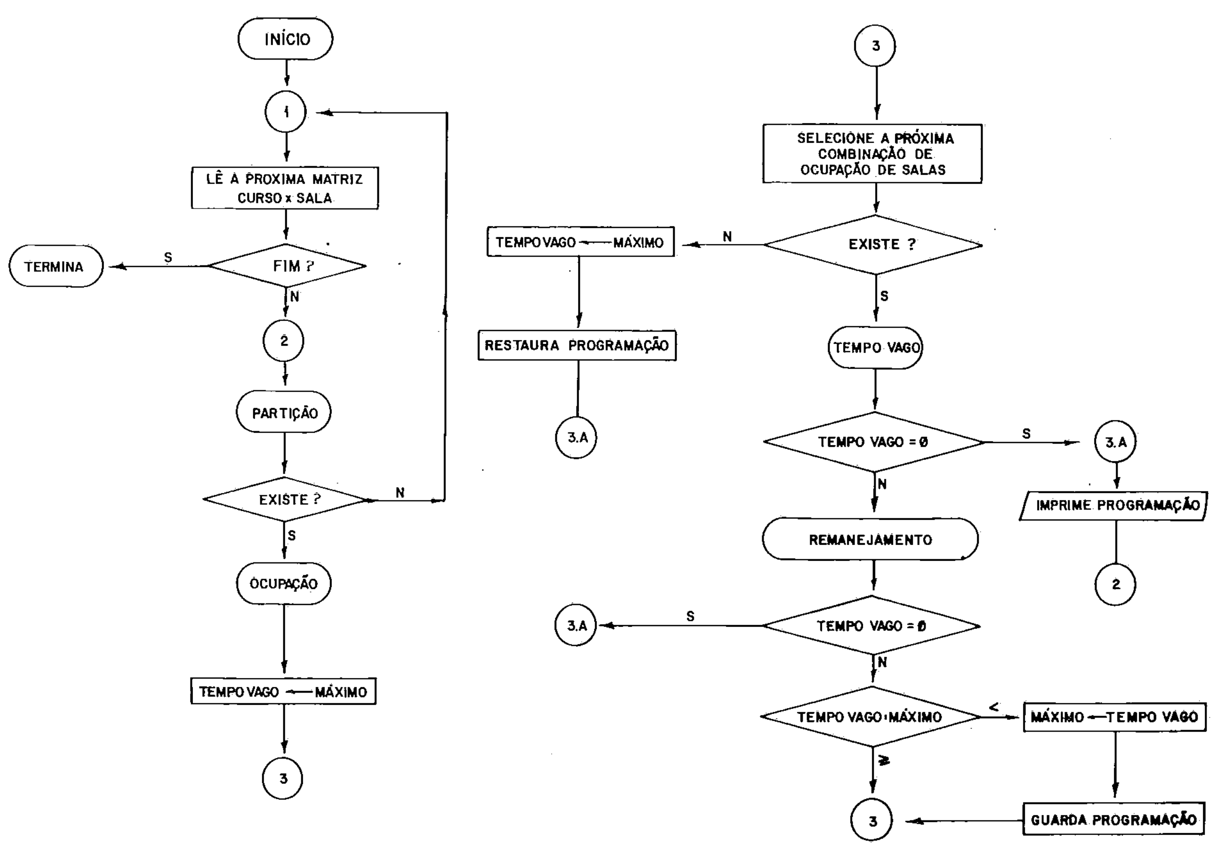


Figura 8
Diagramação de Blocos da Rotina Principal

3.5.2 Rotina: PARTIÇÃO

Determina uma partição pas os cursos não rotulados de uma matriz curso x sala. A partição encontrada é então rotulada. A figura 9 apresenta o diagrama de blocos da rotina.

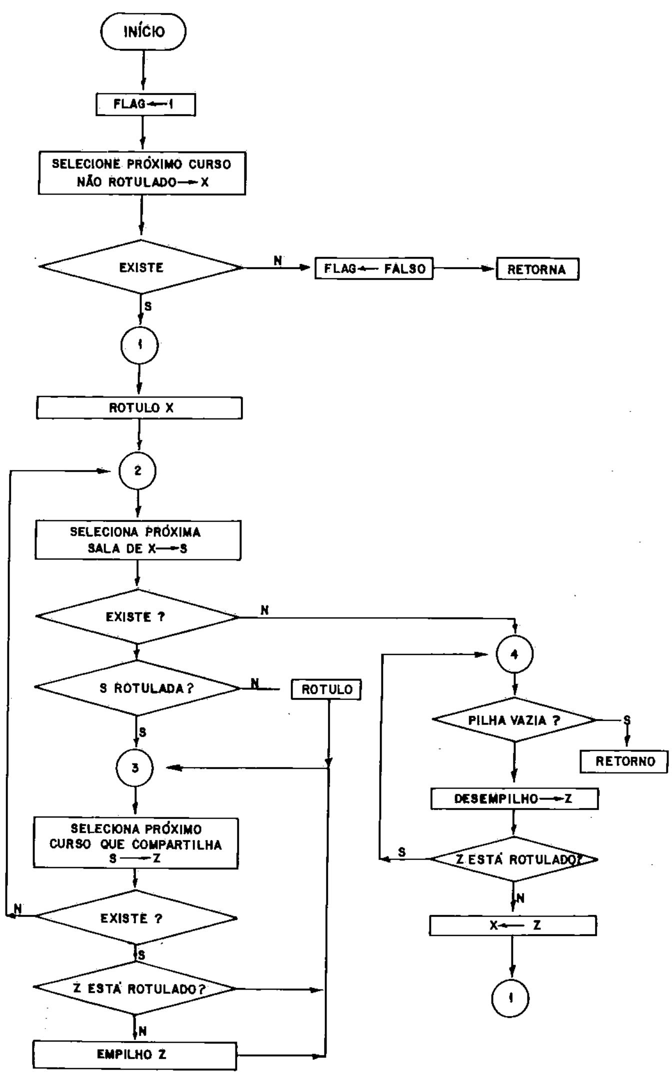


Figura 9
Diagrama de Blocos da Rotina - Partição

3.5.3. Rotina: OCUPAÇÃO

Determina para cada sala o curso preferencial de acordo com os critérios definidos na segunda etapa do algorítmo como mostrado na figura 10.

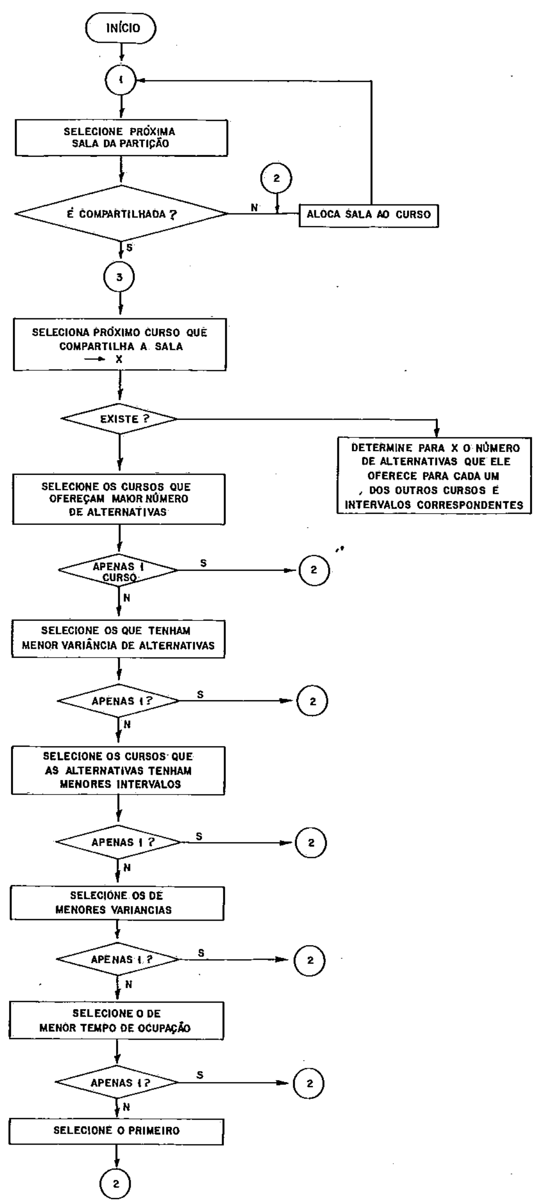
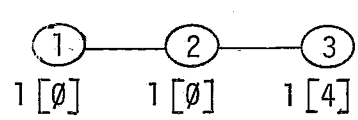


Figura 10
Diagrama de blocos da Rotina - Ocupação

3.5.4. Rotina: TEMPO VAGO

A partir de uma configuração inicial de ocupação das salas,esta rotina determina uma programação de horários que minimize o tempo vago para os cursos. Ela é responsável pela execução da terceira etapa do algorītmo. Seu procedimento corresponderá portanto, à construção da árvore referida naquela etapa.

A construção da árvore segundo a descrição da seção 3.3, exige da rotina a capacidade, denominada aqui de <u>avanço</u> de se definir e rotular os nodos e a capacidade de <u>retroceder</u> (e voltar) do (para) último nodo rotulado da árvore, rearranjando as sequências de modo a obter uma outra árvore, se for possível, com o tempo vago menor e em qualquer caso prosseguir ou de novo retroceder do referido nodo. Esta capacidade pode ser ilustrada pelo exemplo simples da lista encadeada mostrada a seguir.



Suponha que o último curso a ser rotulado foi o 2, obtendo-se o tempo de espera igual a 4 para o curso 3. O procedimento da rotina deverá ser:

encontrar outra sequência para 2 e em seguida para o curso 3 de modo a se obter um tempo menor de que 4. Se for obtido o tempo vago igual a zero então a rotina prosseguirá normalmente. Se o tempo vago for maior do que zero ela, após ter esgotado as possibilidades do curso 2, retrocederá ao curso 1, antes guardando o valor atual do tempo vago e o estado da árvore e selecionará a próxima sequência de 1 e irá avançar de mo do normal até rotular o curso 2 de novo. São então comparados os dois valores do tempo vago escolhendo-se o menor. Se não for zero o procedimento se repete escolhendo agora a sequência 3 (se houver) do curso 1 e assim por diante até que ou se esgote as possibilidades de 1 ou se obtenha o tempo vago igual a zero.

A rotina pode ser então implementada através da organização das informações numa pilha onde cada elemento contém as seguintes informações:

CURSO	SEQ	TEMPO	TEMPO VAGO	RÓTULO	ORIGEM

onde:

CURSO: número do curso

SEQ: número de ordem da sequência

TEMPO: soma dos tempos de espera do CURSO com os tempos de espera dos cursos que compartilham sala com ele.

RÓTULO: se está ou não rotulado

ORIGEM: número do curso que o colocou na pilha

a pilha é controlada pelos ponteiros:

AVANÇO - aponta para o último elemento rotulado

TOPO - aponta para o topo da pilha

são ainda definidas as seguintes variáveis:

TEMPO VAGO - valor do tempo vago corrente

TEMPO VAGO MÍNIMO - valor do tempo vago antes de se retroceder.

MAX - último elemento rotulado antes de retroceder.

VOLTA - indica em qual elemento que se encontra o retrocedimento

FLAG - indica que a rotina se encontra no retrocedimento

G - cópia do estado de programação antes de se retroceder.

MÍNIMO - tempo de espera mínimo obtido no procedimento de rotular um curso.

RETORNO - indica que se encontra no procedimento de rotulação

L - cópia do estado de programação no procedimento de rotulação.

O diagrama de blocos de rotina é apresentado nas figuras 11 e 12.

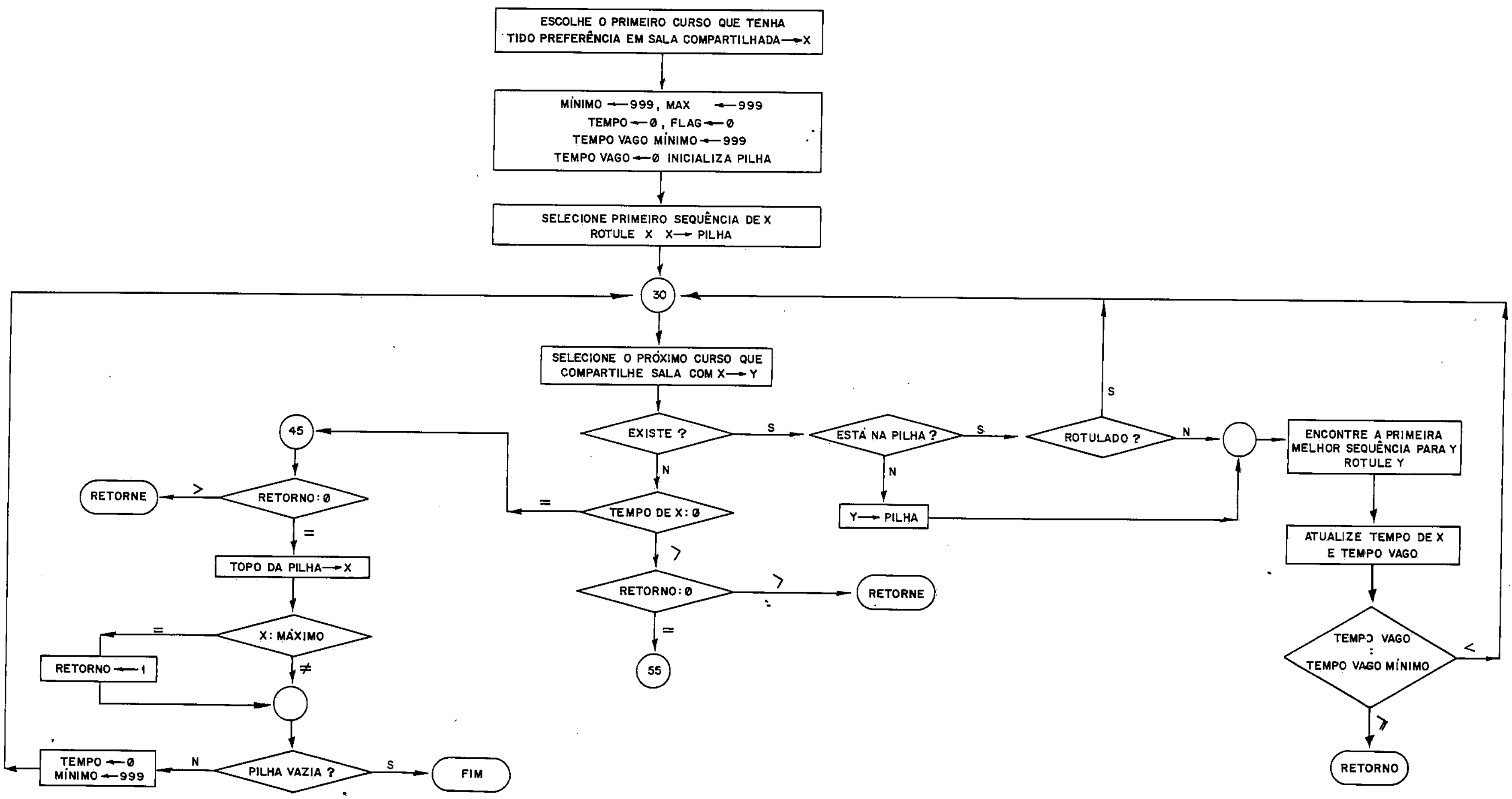


Figura 11

Diagrama de Blocos da Rotina - Tempo Vago

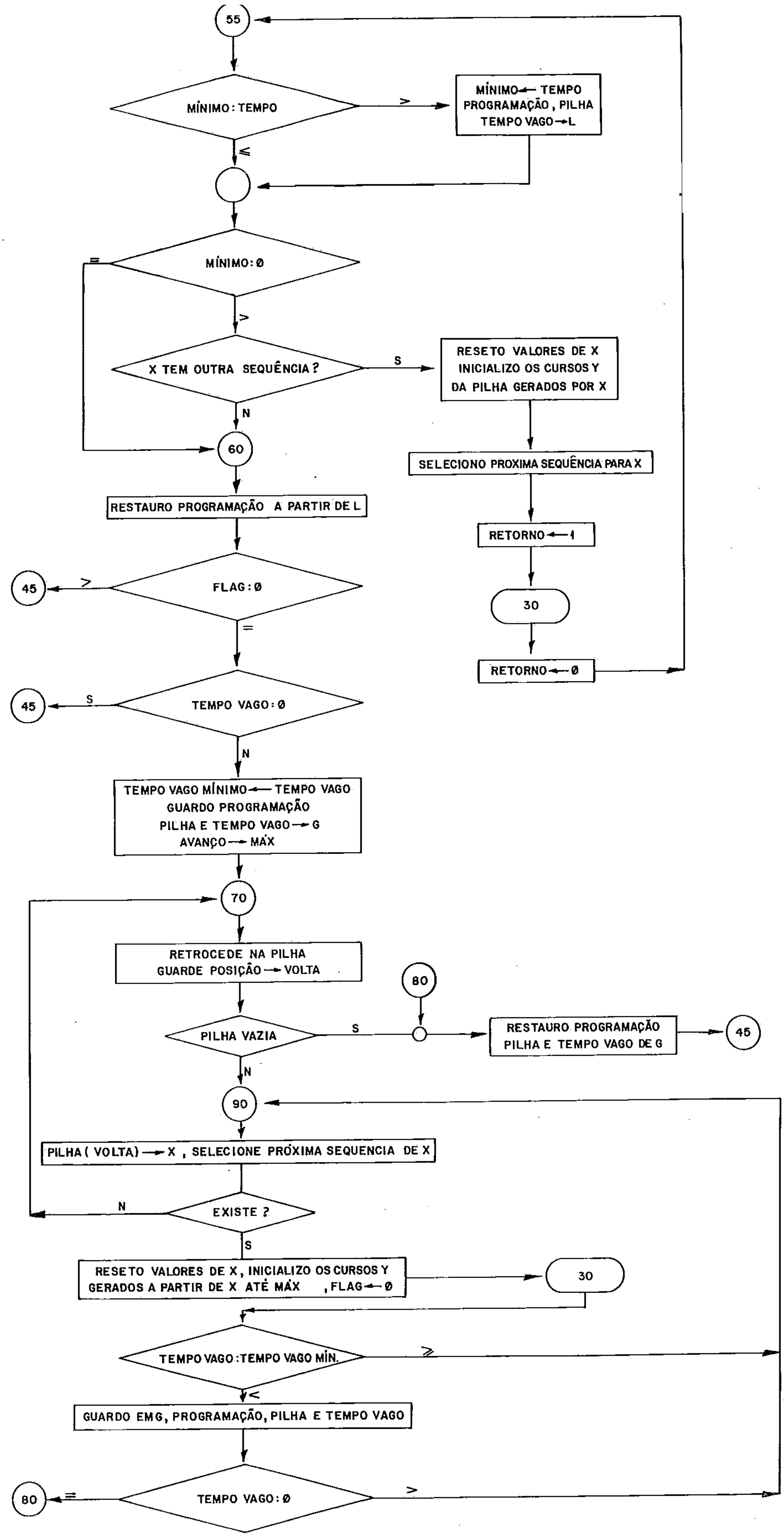


Figura 12

Diagrama de Blocos da Rotina - Tempo Vago (cont.)

CAPÍTULO IV

CONCLUSÕES E SUGESTÕES PARA FUTUROS TRABALHOS

A estratégia adotada viabiliza a resolução do problema de programação de horários em sua forma ampla, que consiste em não obter "simplesmente" um preenchimento dos horários disponíveis com aulas mas sim considerando, o preenchimento dos horários como o resultado de uma série de objetivos a serem atingidos, tais como: otimização dos recursos de salas, uniformidade da carga horária dos cursos, minimização dos horários vagos e as preferências dos professores em relação aos dias de semana; admitindo o tempo e a frequência das aulas variáveis. Sob tal variedade de objetivos, a resolução do problema através de sua decomposição nos sub-problemas de programação das aulas dos cursos nas salas, nos dias da semana e finalmente nos horários disponíveis parece acertada.

Em relação ao tempo de processamento, os valores obtidos da ordem de poucos segundos para amostras de 10 cursos, indicam que os procedimentos heurísticos adotados viáveis.

O ponto de partida do problema correspondendo às informações de sala, tempo, frequência e curso de cada disciplina permitiu colocar o problema sob uma forma simples e aberta de modo a tornar os resultados operacionais. Admite-se também que o método proposto não é uma forma única e acabada de resolução do problema, mas sim uma contribuição para o desenvolvimento de futuros trabalhos. Sob este pressuposto procedem as seguintes sugestões.

No capítulo II a definição de curso implica que os horários das disciplinas de dois cursos são independentes. Assim implicitamente são associadas turmas independentes para cada curso e período letivo. Na prática as turmas existem no ciclo básico das Universidades, enquanto que nos departamentos segundo uma organização "natural", a turma é menos padronizada embora nada indique que não possa existir uma organização mais adequada se as vantagens de um planejamento, forem mais significativas.

O procedimento heurístico deste capítulo admite um aperfeiçoamento do critério de seleção das disciplinas.

Da mesma forma no capítulo III admite-se um aprimoramento do procedimento heurístico para a minimização dos horários vagos, que inclusive, possa resolver o caso geral.

Ainda neste capítulo a fixação dos horários das aulas poderá ser estabelecidos com o objetivo de se tornar constante e atender as preferências dos professores (como objetivo secundário) em relação aos horários das aulas, resolvendo os conflitos eventuais de horários dos professores que dêem aulas em mais de um curso.

Finalmente convém ressaltar que as sugestões acima farão sentido de serem, eventualmente, exploradas após a aplicação do método a problemas de planejamento de horários reais.

A P Ê N D I C E 1

1.1 Teste do algorítmo para a programação de horários nos dias da semana (amostra 1).

1.2 Matrizes curso x sala segundo os dias da semana com as respectivas partições.

1.3 Parâmetros das rotinas.

CURSO: 1

DISCIPL	FREQ	TEMPO	SALA	PRT	SEG	TER	QUA	QUI	SEX	ACUMUL
5	5	7	1	1	7	7	7	7	7	35
6	5	5	2	1	5	5	5	5	5	60
9	5	7	2	1	7	7	7	7	7	95
8	4	5	2	1	5	5	0	5	5	115
2	2	2	1	1	2	0	2	0	0	119
4	2	7	1	3	0	0	7	0	7	133
7	2	8	1	2	0	8	0	8	0	149
1	1	4	1	1	4	0	0	0	0	153
3	1	6	1	3	0	0	6	0	0	159

TOTAL: 30 32 34 32 31
MEDIA= 31.80 VAR= 1.76 PREF. = 1.56

CURSO: 2

DISCIPL	FREQ	TEMPO	SALA	PRT	SEG	TER	QUA	QUI	SEX	ACUMUL
6	4	2	3	1	2	2	0	2	2	8
2	3	5	3	1	5	0	5	0	5	23
3	3	6	3	9	0	6	6	6	0	41
1	2	8	3	6	8	0	0	0	8	57
4	2	4	3	2	0	4	0	4	0	65
5	1	6	3	3	0	0	6	0	0	71
7	1	7	3	2	0	7	0	0	0	78

TOTAL: 15 19 17 12 15
MEDIA= 15.60 VAR= 5.44 PREF.= 3.43

CURSO: 3

DISCIPL	FREQ	TEMPO	SALA	PRT	SEG	TER	QUA	QUI	SEX	ACUMUL
3	4	2	4	1	2	2	0	2	2	8
4	4	4	4	2	4	4	4	0	4	24
5	4	8	4	3	8	0	8	8	8	56
2	3	5	4	9	0	5	5	5	0	71
1	1	8	4	2	0	8	0	0	0	79
6	1	3	5	1	3	0	0	0	0	82
7	1	4	5	5	0	0	0	0	4	86

TOTAL: 17 19 17 15 18
MEDIA= 17.20 VAR= 1.76 PREF.= 3.29

CURSO: 4

DISCIPL	FREQ	TEMPO	SALA	PRT	SEG	TER	QUA	QUI	SEX	ACUMUL
2	5	6	5	1	6	6	6	6	6	30
5	5	2	5	1	2	2	2	2	2	40
7	5	8	6	1	8	8	8	8	8	80
1	4	3	5	1	3	3	0	3	3	92
3	4	2	5	4	2	2	2	2	0	100
6	4	6	6	2	6	6	6	0	6	124
4	1	4	5	3	0	0	4	0	0	128

TOTAL: 27 27 28 21 25
MEDIA= 25.60 VAR= 6.24 PREF.= 1.86

CURSO: 5

DISCIPL	FREQ	TEMPO	SALA	PRT	SEG	TER	QUA	QUI	SEX	ACUMUL
1	5	5	7	1	5	5	5	5	5	25
5	5	2	7	1	2	2	2	2	2	35
3	4	3	6	1	3	3	0	3	3	47
2	2	4	5	1	4	0	4	0	0	55
6	2	5	7	2	0	5	0	5	0	65
4	1	8	7	5	0	0	0	0	8	73

TOTAL: 14 15 11 15 18
MEDIA= 14.60 VAR= 5.04 PREF.= 1.83

CURSO: 6

DISCIPL	FREQ	TEMPO	SALA	PRT	SEG	TER	QUA	QUI	SEX	ACUMUL
2	5	3	7	1	3	3	3	3	3	15
1	4	8	8	1	8	8	0	8	8	47
4	3	3	7	1	3	0	3	0	3	56
5	3	2	8	9	0	2	2	2	0	62
3	2	7	8	1	7	0	7	0	0	76
6	2	4	8	2	0	4	0	4	0	84

TOTAL: 21 17 15 17 14
MEDIA= 16.80 VAR= 5.76 PREF.2=50

CURSO: 7

DISCIPL	FREQ	TEMPO	SALA	PRT	SEG	TER	QUA	QUI	SEX	ACUMUL
3	5	4	9	1	4	4	4	4	4	20
1	3	6	8	1	6	0	6	0	6	38
5	3	4	9	2	4	4	0	4	0	50
2	2	4	9	2	0	4	0	4	0	58
6	2	5	9	3	0	0	5	0	5	68
4	1	5	9	1	5	0	0	0	0	73
7	1	6	9	3	0	0	6	0	0	79
			TOTAL:		19	12	21	12	15	

MEDIA= 15.80 VAR= 13.36 PREF.= 1.86

CURSO: 8

DISCIPL	FREQ	TEMPO	SALA	PRT	SEG	TER	QUA	QUI	SEX	ACUMUL
6	4	6	10	1	6	6	0	6	6	24
1	3	2	9	2	2	2	0	2	0	30
2	3	3	9	1	3	0	3	0	3	39
7	3	3	10	9	0	3	3	3	0	48
3	2	8	10	3	0	0	8	0	8	64
4	2	5	10	2	0	5	0	5	0	74
5	1	4	6	4	0	0	0	4	0	78
			TOTAL:		11	16	14	20	17	

MEDIA= 15.60 VAR= 9.04 PREF.= 3.14

CURSO: 9

DISCIPL	FREQ	TEMPO	SALA	PRT	SEG	TER	QUA	QUI	SEX	ACUMUL
2	5	5	11	1	5	5	5	5	5	25
3	5	2	10	1	2	2	2	2	2	35
5	5	3	11	1	3	3	3	3	3	50
4	2	7	11	1	7	0	7	0	0	64
1	1	8	10	1	8	0	0	0	0	72
6	1	8	11	2	0	8	0	0	0	80
7	1	3	7	3	0	0	3	0	0	83
			TOTAL:		25	18	20	10	10	

MEDIA=16.60 VAR= 34.24 PREF.= 1.43

CURSO: 10

DISCIPL	FREQ	TEMPO	SALA	PRT	SEG	TER	QUA	QUI	SEX	ACUMUL
8	5	6	12	1	6	6	6	6	6	30
4	4	2	12	1	2	2	0	2	2	38
9	4	2	12	2	2	2	2	0	2	46
3	2	8	12	9	0	0	8	8	0	62
1	1	5	11	5	0	0	0	0	5	67
2	1	7	11	4	0	0	0	7	0	74
5	1	2	3	4	0	0	0	2	0	76
6	1	5	11	1	5	0	0	0	0	81
7	1	4	9	5	0	0	0	0	4	85
			TOTAL:		15	10	16	25	19	

MEDIA= 17.00 VAR= 24.40 PREF.= 3.56

Apêndice 1.2 - Matrizes curso x sala segundo os dias da semana com as respectivas partições.

SEG

CURSO	SL1	SL2	SL3	SL4	SL5	SL6	SL7	SL8	SL9	SL10	SL11	SL12	SL
1	13	17	0	0	0	0	0	0	0	0	0	0	30
2	0	0	15	0	0	0	0	0	0	0	0	0	15
3	0	0	0	14	3	0	0	0	0	0	0	0	17
4	0	0	0	0	13	14	0	0	0	0	0	0	27
5	0	0	0	0	4	3	7	0	0	0	0	0	14
6	0	0	0	0	0	0	6	15	0	0	0	0	21
7	0	0	0	0	0	0	0	6	13	0	0	0	19
8	0	0	0	0	0	0	0	0	5	6	0	0	11
9	0	0	0	0	0	0	0	0	0	10	15	0	25
10	0	0	0	0	0	0	0	0	0	0	5	10	15
TOTAL	13	17	15	14	20	17	13	21	18	16	20	10	

TER

CURSO	SL1	SL2	SL3	SL4	SL5	SL6	SL7	SL8	SL9	SL10	SL11	SL12	SL
1	15	17	0	0	0	0	0	0	0	0	0	0	32
2	0	0	19	0	0	0	0	0	0	0	0	0	19
3	0	0	0	19	0	0	0	0	0	0	0	0	19
4	0	0	0	0	13	14	0	0	0	0	0	0	27
5	0	0	0	0	0	3	12	0	0	0	0	0	15
6	0	0	0	0	0	0	3	14	0	0	0	0	17
7	0	0	0	0	0	0	0	0	12	0	0	0	12
8	0	0	0	0	0	0	0	0	2	14	0	0	16
9	0	0	0	0	0	0	0	0	0	2	16	0	18
10	0	0	0	0	0	0	0	0	0	0	0	10	10
TOTAL	15	17	19	19	13	17	15	14	14	16	16	10	

QUA

CURSO	SL1	SL2	SL3	SL4	SL5	SL6	SL7	SL8	SL9	SL10	SL11	SL12	SL
1	22	12	0	0	0	0	0	0	0	0	0	0	34
2	0	0	17	0	0	0	0	0	0	0	0	0	17
3	0	0	0	17	0	0	0	0	0	0	0	0	17
4	0	0	0	0	14	14	0	0	0	0	0	0	28
5	0	0	0	0	4	0	7	0	0	0	0	0	11
6	0	0	0	0	0	0	6	9	0	0	0	0	15
7	0	0	0	0	0	0	0	6	15	0	0	0	21
8	0	0	0	0	0	0	0	0	3	11	0	0	14
9	0	0	0	0	0	0	3	0	0	2	15	0	20
10	0	0	0	0	0	0	0	0	0	0	0	16	16
TOTAL	22	12	17	17	18	14	16	15	18	13	15	16	

QUI

CURSO	SL1	SL2	SL3	SL4	SL5	SL6	SL7	SL8	SL9	SL10	SL11	SL12	SL
1	15	17	0	0	0	0	0	0	0	0	0	0	32
2	0	0	12	0	0	0	0	0	0	0	0	0	12
3	0	0	0	15	0	0	0	0	0	0	0	0	15
4	0	0	0	0	13	8	0	0	0	0	0	0	21
5	0	0	0	0	0	3	12	0	0	0	0	0	15
6	0	0	0	0	0	0	3	14	0	0	0	0	17
7	0	0	0	0	0	0	0	0	12	0	0	0	12
8	0	0	0	0	0	4	0	0	2	14	0	0	20
9	0	0	0	0	0	0	0	0	0	2	8	0	10
10	0	0	2	0	0	0	0	0	0	0	7	16	25
TOTAL	15	17	14	15	13	15	15	14	14	16	15	16	

SEX

CURSO	SL1	SL2	SL3	SL4	SL5	SL6	SL7	SL8	SL9	SL10	SL11	SL12	SL
1	14	17	0	0	0	0	0	0	0	0	0	0	31
2	0	0	15	0	0	0	0	0	0	0	0	0	15
3	0	0	0	14	4	0	0	0	0	0	0	0	18
4	0	0	0	0	11	14	0	0	0	0	0	0	25
5	0	0	0	0	0	3	15	0	0	0	0	0	18
6	0	0	0	0	0	0	6	8	0	0	0	0	14
7	0	0	0	0	0	0	0	6	9	0	0	0	15
8	0	0	0	0	0	0	0	0	3	14	0	0	17
9	0	0	0	0	0	0	0	0	0	2	8	0	10
10	0	0	0	0	0	0	0	0	4	0	5	10	19
TOTAL	14	17	15	14	15	17	21	14	16	16	13	10	

1.3 - Parâmetros das rotinas

1. Rotina: Gera

   Variáveis:

   - FREQ (K): frequência da disciplina K

     $5 \leq FREQ(K) \leq 1$

   - TDISC(K): tempo da disciplina K

     $8 \leq TDISC(K) \leq 2$

   - K : número de ordem da disciplina

     $14 \leq K \leq 4$

   tempo semanal disponível das salas: 80 tempos

2. Rotina: Ordena (OPC, QQ, II, FF, ORD)

   variáveis de entrada

   OPC = 1 → ordem crescente
   OPC = 2 → ordem decrescente

   QQ( ) - vetor fora de ordem

   II - início

   FF - fim

   variáveis de saída:

   ORD( ) - vetor de ponteiros indicadores de ordem dos elementos.

3. Rotina: Alternativas (OPC,FREQ,VALOR,MODOS,MAX,PORD,ALT,NA)

   variáveis de entrada.

   OPC = 1 - gera as alternativas de valor mínimo
   OPC = 2 - gera todas as alternativas

   FREQ - frequência da disciplina

   VALOR ( ) vetor carga horária do curso em cada dia da semana.

MODOS - são as possīveis combinações da frequência FRQ dada.

MAX - nūmero māximo de combinações

variāveis de saīda

AS( ) - vetor alternativas

NS - nūmero de elementos do vetor AS

SORD - ordem do vetor alternativas

4. Rotina: Solução (OPC,FRQ,DISP,XORD,MAX2,VALOR,PRT,MAX1,TORD, SOL, ORD,NA).

variāveis de entrada

FRQ - frequência da disciplina

DISP - conjunto das combinações viāveis (primārio)

XORD - ordem dos elementos de DISP

MAX2 - nūmero de elementos de DISP

VALOR( ) - carga horāria do curso

PRT ( ) - conjunto das combinações preferenciais da disciplina

MAX1 - nūmero de elementos de PRT

TORD - ordem dos elementos de PRT

variāveis de saīda

SOL - vetor solução

ORD - ordem dos elementos de SOL

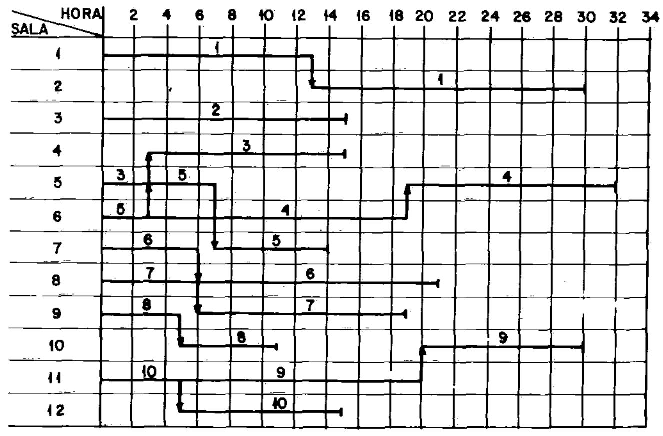
NA - nūmero de elementos de NA

APÊNDICE 2
Programação de horários para a amostra 1

SEGUNDA-FEIRA

CURSO \ SALA	1	2	3	4	5	6	7	8	9	10	11	12
1	13	17										
2			15									
3				14	3							
4					13	14						
5					4	3	7					
6							6	15				
7								6	13			
8									5	6		
9										10	15	
10											5	10

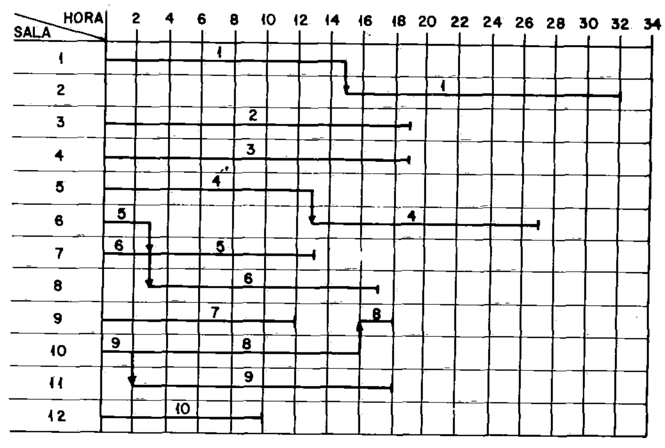
PROGRAMAÇÃO DE HORÁRIOS



TERÇA-FEIRA

CURSO \ SALA	1	2	3	4	5	6	7	8	9	10	11	12
1	15	17										
2			19									
3				19								
4					13	14						
5						3	12					
6							3	14				
7									12			
8									2	14		
9										2	16	
10												10

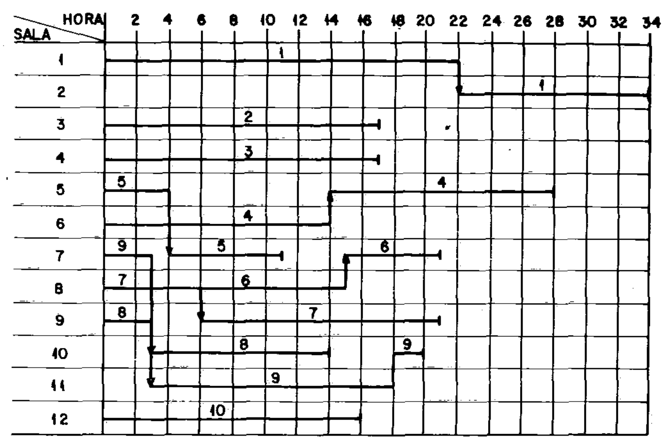
PROGRAMAÇÃO DE HORÁRIOS



QUARTA-FEIRA

CURSO \ SALA	1	2	3	4	5	6	7	8	9	10	11	12
1	22	12										
2			17									
3				17								
4					14	14						
5					4		7					
6							6	9				
7								6	15			
8									3	11		
9							3			2	15	
10												16

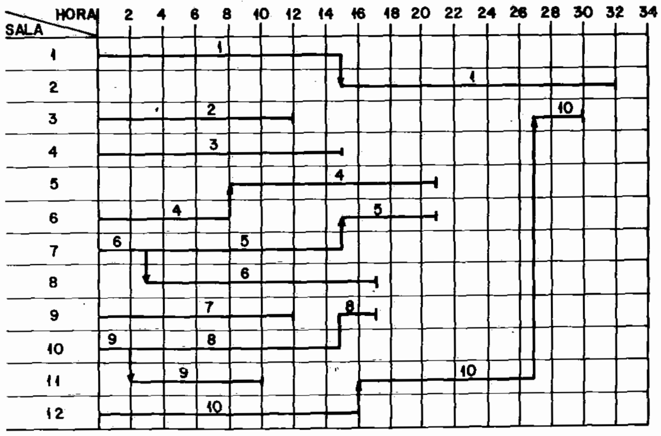
PROGRAMAÇÃO DE HORÁRIOS



QUINTA-FEIRA

CURSO \ SALA	1	2	3	4	5	6	7	8	9	10	11	12
1	15	17										
2			12									
3				15								
4					13	8						
5						3	12					
6							3	14				
7									12			
8						4			2	14		
9										2	8	
10			2								7	16

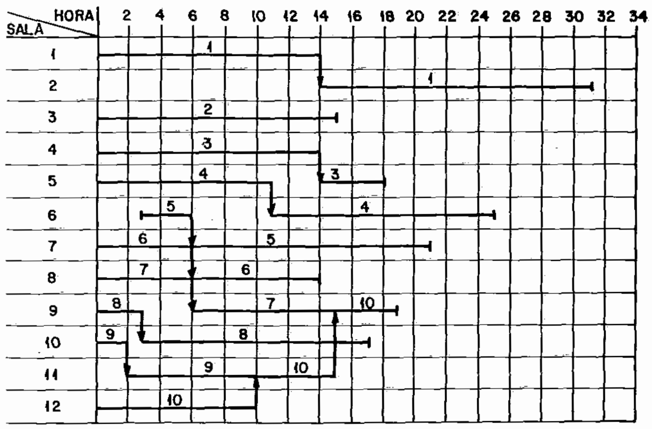
PROGRAMAÇÃO DE HORÁRIOS



SEXTA-FEIRA

CURSO \ SALA	1	2	3	4	5	6	7	8	9	10	11	12
1	14	17										
2			15									
3				14	4							
4					11	14						
5						3	15					
6							6	8				
7								6	9			
8									3	14		
9										2	8	
10									4		5	10

PROGRAMAÇÃO DE HORÁRIOS



B I B L I O G R A F I A

1 APPLEBY, Brake, Newman - Techniques for producing School timetables on a computer and their application to other scheduling problems - The Computer Journal, H3 1961.

2 BOSLER, U e Frangos D. - Strategien bei der Stundenplanerstellung Vergleich und Darstellung der COSMO - strategie-teil 1,2 - 1974.

3 BOSLER, U - Entwichlung eines Programmsystem zur autonatischen Stundenplanerstellung fur Geramtschulen - Dissertation TU - Berlin FB20 - Sommer 1973.

4 CSIMA, J - Investigations on a Timetable Problem - PhD thesis University of Toronto - 1963.

5 CSIMA, J e Gotlieb C.C - A computer Method for Constructing School timetables - Comunication of ACM - 1964.

6 DANTZIG, George B. - Linear Programming and Extensions Princeton University Press, Princeton - NY, 1963.

7 FORD, L. R e Fulkerson D. R - Flows in Networks - Princeton University Press - 1962.

8 GOTLIEB, C.C - The Construction of Class - teacher - timetables Proceedings of the IFIP Congress 62 - North Holland Publ. CO - Amsterdam - 1963.

9 GRAUSS, W - Zur Erstellung eines Horsaalbelegungsplanes- Elektronische Datenverarbeitung 9 - 1968.

10 GRAUSS, W - Ein optimales Zuweisungsschemma für Hörsale

Diplomarbeit 1967 - Univ. München.

[11] LAZAK, D - Ein mathematisches Modell zur Erstellung von Studenplänen - Elektronische Datenverarbeitung 8 1966.

[12] SALKIN, Harvey M - Integer Programming - Addison Wesley -1975

[13] UHLEMANN, K. H et all - Untersuchungen zum Studenplanproblem- Elektronische Datenverarbertung 11 - 1969.

[14] Werra, D - Construction of School timetables by flow methods- INFOR (Canadã) 9 - 1971 March.

[15] Werra, D - Resolution de problemes d'horaire par la thẽorie de graphes - thẽse de doctorat - Ecole Polytechinique Lausanne - 1969.