

Paradigmas da Programação II

LESI + LMCC (1º ano)

Exame de 1ª Época — 1ª Chamada

Data: 23 de Junho de 2004

Hora: 09:30

Versão: A

<p>Dispõe de 2:30 horas para realizar este exame, que tem 7 perguntas V/F (14valores) e 3 perguntas de desenvolvimento (6valores).</p>
<p>Responda na folha fornecida em anexo e não se esqueça de preencher o cabeçalho.</p>
<p>PARA CADA ALÍNEA, INDIQUE SE É (F)alsa ou (V)erdadeira, <u>Critério de Correção:</u> 4 alíneas certas ⇒ 100% 3 alíneas certas ⇒ 50% 2 alíneas certas ⇒ 25%</p>

I.PARTE: Questionário V/F

1 Questão (proc. numérico sequencial)

Observe a função main() escrita na linguagem C como se lista abaixo:

```
int main()
{ int n, ref, d, m=0;
  scanf("%d",&ref);
  scanf("%d",&n);
  while ( d=ref-n ) { m = (d>0)? ref:n; scanf("%d",&n);}
  printf("%d\n",m);
  return( 0 );
}
```

e em relação a ela, analise a veracidade das afirmações seguintes:

- a) O programa termina quando ref for igual a n.
- b) Nada garante que o programa apresentado termine.
- c) O programa escreve o maior valor lido.

d) Se o primeiro número introduzido for 100 e o penúltimo for 50, o programa escreve 100.

%_____

2 Questão (arrays e proc. numérico sequencial)

%_____

3 Questão (strings)

%_____

4 Questão (matrizes)

Para armazenar um horário relativo à ocupação de uma sala (nome da disciplina), das 8h00 às 20h00 e de 2^{af} a Sábado, decidiu-se usar o tipo `tHorario` assim definido:

```
#define MaxH 12
#define MaxD 6
typedef tDisc tHorario[MaxH][MaxD];
```

e em relação a ela, analise a veracidade das afirmações seguintes:

a) Uma possível declaração para `tDisc` seria:

```
#define MaxS 30
typedef char tDisc[MaxS];
```

b) Uma possível declaração para `tDisc` seria:

```
typedef char *tDisc;
```

c) Supondo que `Sala1` era uma variável do tipo `tHorario` devidamente inicializada, para imprimir todas as ocorrências (*dia, hora*) de um determinada disciplina "D1" podia usar-se a função `procDisc("D1", Sala1)` assim declarada:

```
void procDisc(tDisc D, tHorario H)
{ int i,j;
  for( i=0; i<MaxH; i++){
    for( j=0; j<MaxD; j++ ) { if (!strcmp(H[i][j],D)) {printf("(%d,%d)\n",j+2,i+8);} }}
}
```

d) Assumindo que `strdup` copia uma string (alocando espaço antes de copiar), a função:

```
void fun(int D, tHorario H)
{ int i,j;
  char *ant="";
  for( i=0; i<MaxH; i++){
    if (strcmp(H[i][D],ant)) {printf("i => %s\n",i+8); strdup(ant,H[i][D]);} }
}
```

imprime as disciplinas alocadas à sala em causa ao longo das 12h de um determinado dia, não repetindo o nome quando a aula ocupa mais de 1 hora.

%_____

5 Questão (ficheiros)

%_____

6 Questão (polinómios)

%_____

7 Questão (listas)

%_____

II.PARTE: Perguntas de Desenvolvimento

8 Questão (função simples)

Escreva a função principal de um programa que lê uma frase (terminada por '.') e conta todos os separadores encontrados (caracteres que não são nem letra, nem dígito). No fim, escreve cada separador e o número de ocorrências.

%_____

9 Questão (função complexa)

%_____

10 Questão (estrutura de dados)

%_____