

INF 1620 – P1 - 17/09/05	Questão 1
Nome:	
Matrícula:	Turma

Considere as equações de movimento para calcular a posição (s) e velocidade (v) de uma partícula em um determinado instante t , dado sua aceleração a , posição inicial s_0 e velocidade inicial v_0 , de acordo com as fórmulas:

$$s = s_0 + v_0 t + \frac{at^2}{2}$$

$$v = v_0 + at$$

Escreva um programa C completo que capture os valores de s_0 , v_0 , a e t , fornecidos pelo usuário via teclado, e calcule e imprima os valores de s e v . Todos os valores tratados no programa devem ser números de ponto flutuante (`float` ou `double`).

INF 1620 – P1 - 17/09/05	Questão 2
Nome:	
Matrícula:	Turma

Considere uma função `converte_coord` que converte de coordenada polar (raio r e ângulo a) para coordenada cartesiana (abscissa x e ordenada y), de acordo com as fórmulas:

$$x = r.\cos(a)$$

$$y = r.\sin(a)$$

O programa abaixo ilustra o uso dessa função, imprimindo os valores 8.498359 e 49.272488, que representam a coordenada cartesiana calculada para os valores utilizados pelo programa.

```
int main (void) {
    float r = 50, a = 1.4; /*ângulo em radianos*/
    float x, y;
    converte_coord(r, a, &x, &y);
    printf("%f %f\n", x, y);
    return 0;
}
```

Implemente a função `converte_coord` de tal forma que o exemplo acima funcione adequadamente. A solução deve funcionar para quaisquer valores de r e a , não podendo ser particularizada para o programa acima. Utilize as funções `cos` e `sin` definidas na biblioteca padrão de C (`math.h` – ver cabeçalhos na última folha) que calculam o cosseno e o seno de um ângulo em radianos.

INF 1620 – P1 - 17/09/05	Questão 3
Nome:	
Matrícula:	Turma

Implemente uma função que receba como parâmetros dois números inteiros n e k , e calcule o coeficiente binomial $C(n,k)$ correspondente, definido pelo seguinte relação recursiva:

$$C(n,0) = 1 \text{ e } C(n,n) = 1 \quad \text{para } n \geq 0$$

$$C(n,k) = C(n-1,k) + C(n-1,k-1) \quad \text{para } n > k > 0$$

O protótipo dessa função deve ser:

```
int coeficiente (int n, int k);
```

INF 1620 – P1 - 17/09/05	Questão 4
Nome:	
Matrícula:	Turma

Implemente uma função que receba como parâmetros um vetor de números inteiros v e seu comprimento n , e verifique se os elementos do vetor correspondem aos termos de uma PA (progressão aritmética), isto é, se existe um número inteiro k tal que $v[i]$ seja igual a $v[0] + i*k$. Caso os elementos do vetor correspondam aos termos de uma PA, a função deve retornar o valor calculado para k . Por exemplo, se for passado para essa função o vetor $\{2,10,18,26,34\}$, ela deve retornar o valor 8. Caso o vetor não corresponda a uma PA, a função deve retornar 0. Assuma que o vetor sempre terá pelo menos três elementos. Essa função deve ter o seguinte protótipo:

```
int testa_PA (int n, int* v);
```

INF 1620 – P1 - 17/09/05	Questão 5
Nome:	
Matrícula:	Turma

Escreva uma função em C que receba como parâmetro uma string e retorne uma nova string com somente as letras do alfabeto (serão removidos caracteres especiais, espaços, dígitos, etc). Por exemplo, se for passada como parâmetro a cadeia de caracteres "# Mat.: 39838-0 DC", a função deve retornar a cadeia "MatDC". A string passada como parâmetro não pode ser alterada. Essa função deve ter o seguinte protótipo:

```
char* converte (char* s);
```

RASCUNHO

Respostas nesta folha não serão consideradas.

Protótipos de funções que podem ser úteis:

stdio.h:

```
int scanf (char* formato, ...);
int printf (char* formato, ...);
FILE* fopen (char* nome, char* modo);
int fclose (FILE* fp);
int fscanf (FILE* fp, char* formato, ...);
int fprintf (FILE* fp, char* formato, ...);
char* fgets(char* str, int size, FILE* fp);
int sscanf(char* str, char* formato, ...);
```

math.h:

```
double sqrt (double x);
double pow (double x, double exp);
double cos (double radianos);
double sin (double radianos);
```

string.h:

```
int strlen (char* s);
int strcmp (char* s, char *t);
char* strcpy (char* destino, char* fonte);
char* strcat (char* destino, char* fonte);
```

stdlib.h:

```
void* malloc (int nbytes);
void free (void* p);
void qsort (void *vet, int n, int tam, int (*comp) (const void*, const void*));
```

Não separe as folhas deste caderno.