

Nome: _____ **GABARITO** _____*Obs.: A prova pode ser feita a lápis! As funções não devem conter valores fixos!*

1. O que será impresso na tela? Mostre o valor que cada variável assume durante a execução e também circule a resposta; não é para explicar o que cada linha faz.

a) (1,0pt) // Dica: código ASCII para o 'A'=65 'a'=97

```
int main() {
    char str[MAX]="uma frase qualquEr";
    int i;

    for(i=0; str[i]!='\0'; i++){
        if(i==0) {
            if((str[i]>='a' && str[i]<='z'))
                str[i]=str[i]-32;
            continue;
        }
        if(str[i]==' ') {
            i++;
            if(str[i]>='a' && str[i]<='z') {
                str[i]=str[i]-32;
                continue;
            }
        }
        else {
            if(str[i]>='A' && str[i]<='Z')
                str[i]=str[i]+32;
        }
    }
    printf("\n%s",str);
}
```

Uma Frase Qualquer

b) (0,5pt)

```
void funcao(char *buf) {
    int i=0,j=0,len=strlen(buf);
    for(i=0,j=0 ; i<len ; i++) {
        if(buf[i] == ' ') {
            for(j=i ; j<len ; j++)
                buf[j] = buf[j+1];
            len--;
        }
    }
}

main() {
    char str[]="uma frase de teste";
    funcao(str);
    printf("\n%s",str);
}
```

umafraседeteste

c) (1pt)

main() {

```

int vet[5]={1,2,3,4,5};
int i,*pt,x,n=5;
for (i=0,x=0,pt=vet; i < n; i++) {
    x = x + *pt;
    pt++;
}
printf("%d",x);
}

```

15

d) (1pt)

```

int main() {
    char str[100]="carrinho massa";
    int i,j,len,len1;

    for(len=0; str[len]!='\0'; len++);
    len1=0;
    for(i=0; i<(len-len1);) {
        if(str[i]==str[i+1]) {
            for(j=i;j<(len-len1);j++)
                str[j]=str[j+1];
            len1++;
        }
        else
            i++;
    }
    printf("\n%s",str);
    return 0;
}

```

carrinho masa

e) (0,5pt)

```

main() {
    char *p1,*p2;
    p1 = (char *)malloc(25);
    p2 = (char *)malloc(25);
    strcpy(p1,"frase");
    strcpy(p2,"teste");
    strcat(p1,p2);
    printf("%s",p1);
}

```

fraseteste

f) (1pt)

```

int main(void) {
    int i;
    char *substring = "Flum";
    char *frase = "Universidade Federal Fluminense";
    char* resultado = strstr(frase, substring);

    for (i = 0; i < strlen(substring); i++)
        printf("%c", *(resultado + i));

    return 0;
}

```

Flum

2. (3pts) Desenvolva uma função 'divisiveis' que recebe um vetor de inteiros e retorna um novo vetor somente com os números divisíveis por um número (*num*) recebido como parâmetro. A função deve ser

genérica, para qualquer tamanho de vetor. Na main, chame a função 'divisiveis' e imprima o vetor resultante. Teste com os divisíveis de 5 para o vetor[6]={1,2,5,10,20,24};

O protótipo da função deve ser:

```
int *divisiveis(int *vetor, int n, int num, int *tam);
```

#define N 6

```
int *divisiveis(int *vetor, int n, int num, int *tam) {
    int i, cont=0, j=0;
    for (i=0; i < n; i++)
        if (vetor[i]%num==0)
            cont++;
    *tam = cont;
    int *divisiveis=(int *)malloc(cont*sizeof(n));

    for (i=0,j=0;i<n;i++)
        if (vetor[i]%num==0)
            divisiveis[j++]=vetor[i];

    return divisiveis;
}
```

```
main(void) {
    int i=0,vetor[N]={1,2,5,10,20,24};
    int *divPor5, tamNovo;
    divPor5 = divisiveis(vetor,N,5,&tamNovo);
    for (i=0;i<tamNovo;i++)
        printf("%d ",divPor5[i]);
}
```

3. (2pts) Desenvolva um programa que declare um double e o inicialize com qualquer valor. Posteriormente, passe essa variável por referência para uma função que dobra o seu valor. Imprima o novo valor na main.

```
void funcao(double *ptr) {
    *ptr = *ptr * 2;
}
```

```
main(void) {
    double x=2;
    funcao (&x);
    printf("%lf",x);
}
```