Prova de Linguagem de Programação, 2o ADS- B-Matutino

(1)(1,0) Qual a diferença de passagem por valor e passagem por referência em uma função?

Em uma função, um parâmetro de passagem de valor, recebe uma cópia do valor do parâmetro, ou seja, a variável original que foi utilizada como parâmetro na função, não será alterada.

No caso da passagem por referência, o parâmetro será uma referência ao endereço da variável original (como um ponteiro), e ao alterar o valor receido dentro da função, a variável original é alterada também

(2)(2,0)Crie uma função em C que tem como parâmetros um vetor e retorne o maior e o menor número deste vetor. Função MaiorMenorValor. Dê exemplos de utilização deste código.

#include <stdio.h>

Int main(){

Int maior=0, menor=99999, vetor[100], n;

For(int i=0; i<100; i++){

Scanf(“%d”, &n);

If(n==0) break;

Else vetor[i] = n;

}

MaiorMenorValor(vetor, &maior, &menor);

Printf(“Maior valor: %d\n”, \*maior);

Printf(“Menor valor: %d\n”, \*menor);

Return 0;

}

void MaiorMenorValor(int vetor[100], int \*maior, int \*menor){

for(int i=0; i<sizeof(vetor); i++){

if(\*maior < vetor[i]) \*maior = vetor[i];

if(\*menor > vetor[i]) \*menor = vetor[i];

}

}

(3)(3,0) Crie uma estrutura que permita manipular números complexos! Considere que nesta estrutura complexo teremos real(int) e imag(int).

Crie uma função que some duas estruturas complexo e retorne o resultado delas. complexo soma(complexo A, complexo B) a + bi + c + di = (a + c) + (b + d)i Crie uma função que multiplique dois números complexos

complexo multi(complexo A, complexo B) (a + bi) ∗ (c + di) = (a ∗ c − b ∗ d) + (b ∗ c + a ∗ d)i Crie uma função que imprima um complexo na forma a+bi.

typedef struct{

int real, imag;

}complexos;

complexos soma(complexos A, complexos B);

complexos multi(complexos A, complexos B);

void imprime(complexos A);

int main(){

complexos A, B;

printf(“Insira a parte real do primeiro numero: ”);

scanf("%d", &A.real);

printf(“Insira a parte imaginaria do primeiro numero: ”);

scanf("%d", &A.imag);

printf(“Insira a parte real do segundo numero: ”);

scanf("%d", &B.real);

printf(“Insira a parte imaginaria do segundo numero: ”);

scanf("%d", &B.imag);

printf(“Soma dos numeros: ”);

imprime(soma(A, B));

printf(“Multiplicacao dos numeros”);

imprime(multi(A, B));

}

complexos soma(complexos A, complexos B){

complexos Soma;

Soma.real = A.real + B.real;

Soma.imag = A.imag + B.imag;

return Soma;

}

complexos multi(complexos A, complexos B){

complexos Mult;

Mult.real = A.real \* B.real;

Mult.imag = A.imag \* B.imag;

return Mult;

}

void imprime(complexos A){

printf("Numero complexo: (%d + %d i)", a.real, a.imag);

}

(4)(2,0) Faça um programa em C que pela linha de comando faça a seguinte função: Leia um arquivo e capture todas as consoantes do arquivo e exibe na tela.

Exemplo: conteúdo do arquivo: tricolor C:\ > ComeVogal arquivo.txt

int main(int argc, char \*argv[]){

FILE \*A;

int c;

if(argc == 2){

A = fopen(argv[1], "r");

if(A!=NULL){

while((c = getc(A)) != EOF){

if(((c = getc(A)) != "a") && (c = getc(A)) != "e") && (c = getc(A)) != "i") && (c = getc(A)) != "o") && (c = getc(A)) != "u")){

printf(“%c”, c);

}

}

}

else printf("Erro na abertura do arquivo!!");

}else printf("Uso da ferramenta: ComeVogal <arquivo.txt>");

return 0;

}

(5) (1,0) Dadas as declarações abaixo; qual é o valor dos itens: int x=10, \*px=&x;



float y = 9.9 , \*py=&y ;

• a) x= 10

• b) \*px= 10

• c) px= FFA0

• d) &y = FFB4

• e) \*px= 10

• f) py= FFB4

(6) (1,0) Seja a sequência de instruções em um programa C. int \*pti;

int i=10;

pti=&i;

Justifique cada uma das afirmações como verdadeira ou falsa

• a) pti armazena o endereço de i

• b) \*pti é igual a 10

• c) ao se executar \*pti=20; i passará a ter o valor 20

• d) ao se alterar o valor de i, \*pti será modificado

• e) pti é igual a 10