TÉCNICO DE ENSINO: Rick Willian Kanashiro

UNIDADE CURRICULAR: Lógica de Programação

CURSO: Técnico em Desenvolvimento de Sistemas

1. Escreva um algoritmo que após o usuário informar um número maior que zero realize a soma de todos os números de 1 até o informado, e que a informe ao final. Faça o programa utilizando as três formas aprendidas, for, while e do...while.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

#include <stdio.h>

int main()

{

int n;

int soma= 0;

int cont= 0;

printf("digite qualquer número maior que 0 ");

scanf("%d",&n);

while(n<=0){

printf("erro, digite um número valido ");

scanf("%d", &n);

}

while(cont<=n){

printf("\n cont vale %d\n",cont);

soma=soma + cont;

printf("soma é %d\n",soma);

cont=cont + 1;

}

printf("\n o valor é igual a %d ", soma);

}

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

**int main()**

**{ int num;**

**int cont1;**

**int soma = 0;**

**do**

**{**

**printf("Digite um número maior que zero: \n");**

**scanf("%d", &num);**

**}while(num<0);**

**for (cont1 = 1;cont1 <= num;cont1++)**

**soma = soma + cont1;**

**{**

**printf("A soma dos números é %d: ", soma);**

**}**

**return 0;**

**}**

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

#include <stdio.h>

int main()

{

int n;

int soma=0;

int cont=0;

do{

printf("digite qualquer numero maior que zero:");

scanf("%d", &n);

}

while(n<0);

do{

soma=cont+ soma;

cont=cont+ 1;

}while(cont<=n);

printf("\n o valor é %d", soma);

}

1. Escreva um algoritmo que após o usuário informar qualquer número diferente de zero mostre a tabuada deste.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

#include <stdio.h>

int main()

{

int n; //valor do usuario

int multi=1; //multiplicador

int res ;// resultado

do{

printf("\*\* TABUADA SUPER LEGAL \*\*");

printf("\n digite um número maior que zero");

scanf("%d", &n);

}while(n<0);

for(multi=1;multi<=10;multi++){

res=n \* multi;

printf("\n o resultado é %d", res);

}

}

1. A sequência de Fibonacci é feita conforme a seguinte sequência: 1, 1, 2, 3, 5... de forma que cada novo número corresponde a soma dos dois últimos números na sequência. Com esse conhecimento, monte um algoritmo mostre o resultado após um usuário digitar quantos números ele deseja obter dessa sequência.

#include <stdio.h>

int main()

{

int n;

int cont=1;

int result;

int atual=1;

int ex=1;

do{

printf(" SEQUENCIA DE FIBONACCI ");

printf("\n digite quantos números deseja obter: ");

scanf("%d", &n);

}while(n<0);

printf("\n%d,%d,",ex, atual);

for(cont=1;cont <n;cont++)

{

result=atual+ex;

printf("%d,", result);

ex=atual;

atual=result;

}

}

1. Um aluno estava estudando para a prova de probabilidade e deparou-se com problemas de permutação, onde 7 pessoas formariam uma fila e era questionado de quantas formas possíveis essa fila seria formada. Após fazer os cálculos, ficou em dúvida em relação a sua resposta, já que se trata de um processo trabalhoso, e pediu para você um código que calculasse a permutação após colocar o número de pessoas desejado.

#include <stdio.h>

int main()

{

int n;

int result=1;

int cont=1;

printf("digite o numero de elementos: ");

scanf("%d", &n);

for(cont=1;n>=1;n--){

result=result \* n;

}

printf("permutação: %d",result);

}

1. Continuando o exercício anterior, é normal após permutação o estudo de combinação, então faça o mesmo lembrando que o n representa a quantidade de elementos ao todo e p quantos serão escolhidos.

#include <stdio.h>

int main()

{

int n;

int p;

int fat=1;

int fat2=1;

int fat3=1;

int result;

int result2;

float result3;

int cont1=1;

int cont2=1;

int cont3=1;

printf("digite o primeiro numero de elementos(n): ");

scanf("%d", &n);

printf("\n digite o segundo numero de elementos(p): ");

scanf("%d", &p);

for(cont1=1;cont1 <= n;cont1++){

fat=fat \* cont1;

}

for(cont2=1;cont2<=p;cont2++){

fat2=fat2 \* cont2;

}

result= n - p;

for(cont3=1;result>=1;result--){

fat3=fat3 \* result;

}

result2= fat3\*fat2;

result3= fat / result2;

printf("\n RESULTADO %f",result3);

}

1. Construa um triângulo da seguinte forma após o usuário passar a quantia de linhas desejada:

\*

##

\*\*\*

####

\*\*\*\*\*

1. Na primeira aula, discutimos sobre os tipos de dados, onde cada tipo possui uma faixa limitada de valores. Dessa forma, utilizando somente a biblioteca padrão faça um algoritmo que retorne os limites dos tipos de dados char, unsigned char e signed char.

#include <stdio.h>

int main()

{

char a =1;

unsigned char b =1;

signed char c =1;

while (a>0)

{

a++;

}

a=a-1;

printf("\no char é %d",a);

while(b>0){

b++;

}

b--;

printf("\no unsigned char é %d",b);

while(c>0){

c++;

}

c--;

printf("\no signed char é %d",c);

}

1. Considere que será necessário somar dois elementos do tipo char, porém a soma pode chegar até a 511. Sabendo que essa quantia ultrapassa o tipo char, e só será necessário mostrar na tela, monte um algoritmo que mostre na tela essa soma utilizando dois tipos char para receber os valores
2. Aproveitando que estudamos os limites da variável char, é desejado montar um programa que realize o calculo da quantia possível de dados, ou seja, se for um tipo de 8bits (como o char), responde aos 256. Tente avaliar os limites do tipo int e unsigned int modificando quantos bits será calculado e o tipo da variável soma no algoritmo criado. Caso necessário, utilize o tipo long ao invés de int, e para exibir utilize %ld (LD).