| L logo 25 anos.png  **Universidade Luterana do Brasil**  **ULBRA – Campus Torres**  **Pró-Reitoria de Graduação** | | Tipo de atividade:  Prova ( ) Trabalho ( ) ..... ( )  Avaliação: G1 ( ) G2 ( )  Substituição de Grau: G1 ( ) G2 ( ) | |
| --- | --- | --- | --- |
| Curso: | Disciplina: | | Data: 22/06/22 |
| Turma: | Professor(a): | | Valor da Avaliação:  Nota: |
| Acadêmico(a): Eduardo Augusto Ribeiro dos Santos n°: | | |

Atividade em Dupla.

Peso 1

1 — Preencha um vetor com 100 números inteiros de forma randômica. Quando o número randômico for 999 o preenchimento deve ser encerrado.

Após preenchido O Programa deve possibilitar 4 opções:  
1- exibir uma lista os números presentes no vetor, não incluído o 999.

2- imprimir o vetor na ordem inversa.

3- Exibir o maior número

4 - Exibir quantos números foram sorteados.

5- Encerrar o programa.

| #include <stdio.h> #include <math.h> #include <locale.h> #include <stdlib.h> #include <time.h>  int main(){    int num[100];  int i=0,cont=0, locEsp=0, opc;  int mai=0;  srand(time(NULL));  setlocale(LC\_ALL, "Portuguese");    for(i=0;i<100;i++){  num[i]=(rand()%1000);  cont+=1;  if(num[i]==999){  locEsp=1;  printf("999 detectado\n");  break;  }  }      while(opc!=5){  if(locEsp=1){  printf("Escolha uma opção:\n");  printf("1-Exibir vetor\n2-Exibir vetor reverso\n3-Exibir maior número\n4-Exibir quantia de números\n5-Encerrar\n");  scanf("%d%\*c", &opc);  } else if (locEsp=0){  pritnf("999 não encontrado, escolha uma opção:\n");  printf("1-Exibir vetor\n2-Exibir vetor reverso\n3-Exibir maior número\n4-Exibir quantia de números\n5-Encerrar\n");  scanf("%d%\*c", &opc);  }  switch (opc){  case 1:  for(i=0;i<cont-1;i++){  printf("%d\t", num[i]);  }  printf("\n");  break;    case 2:  for(i=cont-1;i>=0;i--){  printf("%d\t", num[i]);  }  printf("\n");  break;    case 3:  for(i=0;i<cont;i++){  if(num[i]>mai){  mai=num[i];  }  }  printf("O maior número do vetor é %d\n", mai);  break;    case 4:  printf("Foram sorteados %d números para achar o '999' \n", cont);  break;    case 5:  printf("Encerrando...\n");  break;     default:  printf("Opção não suportada\n");  break;  }   }        return 0; } |
| --- |

2-Faça um algoritmo que Preencha 4 vetores:

* o primeiro com a nota da primeira prova;
* o segundo com a nota da segunda prova;
* o terceiro com as médias ponderada das 2 primeiras notas;
* O Quarto com o nome do Aluno.

O programa deve ao fim de cada leitura perguntar se o usuário deseja adicionar um novo aluno. Defina o vetor com no máximo 10 alunos, Após adicionados:

O Programa deve possibilitar 4 opções:  
1- Listar todos os alunos aprovados

2 -Listar todos os alunos reprovados

3- Listar todos os alunos com nota maior que uma nota informada pelo usuário.

4-Informar uma Posição e o programa exibir os dados e notas do aluno.

5- Encerrar o programa.

| #include <stdio.h> #include <math.h> #include <locale.h>  int main(){    int N1[10], N2[10], M[10];  char Nome[10][40];  int i=0, opc=0, alt=0, lac=0;  int alu=1, misc=1;  int X;      setlocale(LC\_ALL, "Portuguese");    do{  for(i=lac;i<alu;i++){  printf("Insira a primeira nota do aluno %d:\n", misc);  scanf("%d%\*c", &N1[i]);  printf("Insira a segunda nota do aluno %d:\n", misc);  scanf("%d%\*c", &N2[i]);  M[i]=(N1[i]+N2[i])/2;  printf("Insira o nome do aluno %d:\n", misc);  scanf("%s%\*c", &Nome[i]);  printf("A média de %s é %d\n", Nome[i], M[i]);    printf("Gostaria de adicionar mais um aluno? (%d / 10)\n", misc);  printf("0-Sim\t1-Não\n");  scanf("%d%\*c", &alt);  }  if(alt==0){  alu+=1;  misc+=1;  lac+=1;  }else{  opc=1;  break;  }    }while(opc!=1);      while(opc!=5){    printf("Escolha uma opção:\n");  printf("1-Listar alunos aprovados\n2-Listar alunos reprovados\n3-Listar alunos com nota inferior a 'X'\n4-Dados do aluno na posição 'X'\n5-Encerrar\n");  scanf("%d%\*c", &opc);    switch (opc){  case 1:  for(i=0;i<alu;i++){  if(M[i]>7){  printf("\nO aluno %s está aprovado (Nota %d)\n", Nome[i],M[i]);  }  }  printf("\n");  break;    case 2:  for(i=0;i<alu;i++){  if(M[i]<7){  printf("\nO aluno %s está reprovado (Nota %d)\n", Nome[i],M[i]);  }  }  printf("\n");  break;    case 3:  printf("Insira um valor para 'X'\n");  scanf("%d%\*c", &X);  for(i=0;i<alu;i++){  if(M[i]<X){  printf("\nO aluno %s tem nota abaixo de %d (Nota %d)\n", Nome[i], X, M[i]);  }  }  printf("\n");    break;    case 4:  printf("Insira um valor para 'X'\n");  scanf("%d%\*c", &X);  for(i=0;i<alu;i++){  printf("O aluno %s está na posição %d do vetor\n", Nome[i], X);  printf("\n");  }    break;    case 5:  printf("Encerrando...\n");  break;     default:  printf("Opção não suportada\n");  break;  }   }      return 0; } |
| --- |

3 - Fazer um algorítimo que seja lido uma palavra e imprima:

* A quantidade de vogais que compõe a palavra.
* A quantidade de letras dessa palavra.

| #include <stdio.h> #include <math.h> #include <locale.h>  int main(){    char Palavra[42];  int i=0, cont=0;  int misc;    setlocale(LC\_ALL, "Portuguese");    printf("Insira uma palavra: (Até 40 caracteres)\n");  gets(Palavra);    misc = strlen(Palavra);    for(i=0;i<misc;i++){  if(Palavra[i]=='a'||Palavra[i]=='A'||Palavra[i]=='e'||Palavra[i]=='E'||Palavra[i]=='i'||Palavra[i]=='I'||Palavra[i]=='o'||Palavra[i]=='O'||Palavra[i]=='u'||Palavra[i]=='U'){  cont+=1;  }  }    printf("%s possui %d vogais\n", Palavra, cont);  printf("%s possui %d caracteres\n", Palavra, misc);        return 0; } |
| --- |