//incluiu as biblitecas

#include <stdio.h>

#include <stdlib.h>

//definiu os void... Aloca criar um ponteiro para ponteiro. leitura faz somente a leitura do conteudo do ponteiro.

//Media é semelhante o aloca, porem ele so utiliza o conteudo do ponteiro p e cria um novo ponteiro para ponteiro e a sua quantidade;

//mostra é responsavel por mostrar o conteudo do ponteiro e o tamanho;

void aloca (float \*\*p, int tam);

void leitura (float \*p);

void media (float \*p, int tam, float \*\*pm, int \*qm);

void mostra(float \*p, int tam);

//A main é composta pelos ponteiros originais, que devem ser iniciados =NULL, um int cont e um qmenor;

//na main é possivel notar uma estrutura de repetição do-while, que permite com que o usuario va digitando os valores até onde ele quiser.

//depois que ele encerra isso, o programa mostra os valores que foram digitados, a media dele e os que ficaram abaixo da media;

// !!!!NA MAIN SÓ FAZ O USO DAS VARIAVEIS CITADAS NELA!!!!

main()

{

float \*ptr=NULL, \*pmenor=NULL;

int cont=0, qmenor=0;

char op;

do{

aloca(&ptr, cont+1);

leitura(ptr+cont);

cont++;

printf("\nDeseja continuar? (S/N)\n");

scanf("%c", &op);

fflush(stdin);

}while(op!='n' && op!='N');

printf("\nNumeros digitados: \n");

media(ptr, cont, &pmenor, &qmenor);

printf("\nMenores que a media:\n");

mostra(pmenor,qmenor);

system ("pause");

}

//o void aloca faz o uso do realloc, basta decorar a seguinte sintaxe:

//if((\*ponteiroparaponteiro=(tipodoponteiro\*)realloc(\*ponteiroparaponteiro,tamanho\*sizeof(tipodoespaço)))==NULL)

//exit(1) encerra o programa caso a memoria do pc estiver cheia.

void aloca (float \*\*p, int tam){

if((\*p=(float\*)realloc(\*p, tam\*sizeof(float)))==NULL)

exit(1);

}

void leitura (float \*p){

printf("Numero: ");

scanf("%f", p);

fflush(stdin);

}

void media (float \*p, int tam, float \*\*pm, int \*qm){

int i;

float m;

for(i=0;i<tam; i++){

m+=\*(p+i);

m/=tam;

printf("\nMedia = %.2f", m);

}

for(i=0;i<tam; i++){

if(\*(p+i)<m){

aloca(pm,\*qm+1);

\*(\*pm+\*qm)=\*(p+i);

(\*qm)++;

}

}

}

void mostra(float \*p, int tam){

int i;

for(i=0;i<tam;i++){

printf("\n%.2f", \*(p+i));

}

}