141.Crie uma função que recebe uma string e um caractere e apague todas as ocorrências desse caractere na string.(cadeia de caractere).

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

void excluindo\_caractere(char \* palavra,char \* excluido,char carac)

{

int i=0,tam\_palavra=strlen(palavra),k=0;

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

if(palavra[i]==carac){

continue;

}

else{

excluido[k]=palavra[i];

k++;

}

}

}

int main(void)

{

int tam\_palavra;

char \* palavra;

char \* excluido;

char carac;

printf("Tamanho da sua palavra: ");

scanf("%d\n",&tam\_palavra);

palavra=(char \*)malloc((tam\_palavra+1)\*sizeof (char));

excluido=(char \*)malloc((tam\_palavra+1)\*sizeof (char));

scanf("%60[^\n]\n",palavra);

scanf("%c",&carac);

excluindo\_caractere(palavra,excluido,carac);

printf("Retiramos o elemento %c e ficou %s",carac,excluido);

free(palavra);

free(excluido);

return 0;

}

142.Crie uma função que mostra os caracteres repetidos de uma string.

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

void apaga\_caractere(char \* palavra,char \* distinto,char \* repetido)

{

int i,tam\_palavra=strlen(palavra),contDistinto=0,k,nao\_esta\_incluso,s=-1;

distinto[0]=palavra[0];

for(i=1;i<tam\_palavra;i++){

/\*vamos verificar se o elemento palavra[i] está ou não no distinto\*/

nao\_esta\_incluso=1;

for(k=0;k<=contDistinto;k++){

if(palavra[i]==distinto[k]){

s++;

repetido[s]=palavra[i];

nao\_esta\_incluso=0;

break;

}

else{

continue;

}

}

if(nao\_esta\_incluso){

contDistinto++;

distinto[contDistinto]=palavra[i];

}

}

}

int main(void)

{

char \* palavra;

char \* distinto;

char \* repetido;

int n;

printf("quantas letras tem sua palavra:");

scanf("%d\n",&n);

palavra=(char \*)malloc((n+1)\*sizeof(char));

distinto=(char \*)malloc((n+1)\*sizeof(char));

repetido=(char \*)malloc((n+1)\*sizeof(char));

scanf(" %[^\n]",palavra);

apaga\_caractere(palavra,distinto,repetido);

printf("repetido: %s",repetido);

free(palavra);

free(distinto;

free(repetido);

return 0;

}

143.Crie uma função que retira todos os caracteres repetidos de uma string.

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

void apaga\_caractere(char \* palavra,char \* distinto)

{

int i,tam\_palavra=strlen(palavra),contDistinto=0,k,nao\_esta\_incluso;

distinto[0]=palavra[0];

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

/\*vamos verificar se o elemento palavra[i] está ou não no distinto\*/

nao\_esta\_incluso=1;

for(k=0;k<=contDistinto;k++){

if(palavra[i]==distinto[k]){

nao\_esta\_incluso=0;

break;

}

else{

continue;

}

}

if(nao\_esta\_incluso){

contDistinto++;

distinto[contDistinto]=palavra[i];

}

}

}

int main(void)

{

char \* palavra;

char \* distinto;

int n;

printf("quantas letras tem sua palavra:");

scanf("%d\n",&n);

palavra=(char \*)malloc((n+1)\*sizeof(char));

distinto=(char \*)malloc((n+1)\*sizeof(char));

scanf(" %[^\n]",palavra);

apaga\_caractere(palavra,distinto);

printf("distinto: %s",distinto);

free(palavra);

free(distinto);

return 0;

}