

PD2 LAB4 SORULARI

1. Toplam eleman sayısını kullanıcıdan alarak bu sayı kadar bellekten calloc fonksiyonunu kullanarak gerekli yeri alınız. Ayırdığınız bu bellek alanına eleman sayısı kadar değeri kullanıcıdan alınız. Daha sonra girilen en büyük değeri tanımladığınız işaretçi değişkeni kullanarak bulunuz ve ekrana yazdırınız.

By taking the total number of elements from the user, take the necessary place by using the calloc function from memory. The number of elements in this memory area you allocate is taken from the user. Then find the largest value entered using the pointer variable you have defined and print it on the screen.

Çözüm:

```
#include <stdio.h>
#include <stdlib.h>
int main(){
    int i, num;
    float *data;
    printf("Enter total number of elements(1 to 100): ");
    scanf("%d", &num);
    // Allocates the memory for 'num' elements.
    data = (float*) calloc(num, sizeof(float));
    if(data == NULL){
        printf("Error!!! memory not allocated.");
        exit(0);
    }
    printf("\n");
    // Stores the number entered by the user.
    for(i = 0; i < num; ++i){
        printf("Enter Number %d: ", i + 1);
        scanf("%f", data + i);
    }
    // Loop to store largest number at address data
    for(i = 1; i < num; ++i){
        // Change < to > if you want to find the smallest number
        if(*data < *(data + i))
            *data = *(data + i);
    }
    printf("Largest element = %.2f", *data);
    return 0;
```

```
}
```

2. Adınız soyadınız için gerekli minimum char bellek alanını malloc fonksiyonu kullanarak ayırınız ve içerisine strcpy fonksiyonunu kullanarak adınızı ve soyadınızı arada boşluk bırakarak yazınız.

Allocate the minimum char memory space required for your name and surname using the malloc function. Then use the strcpy function to write your name and surname with spaces to this variable.

Çözüm:

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

int main(){
    char *mem_allocation;
    /* memory is allocated dynamically */
    mem_allocation = (char*)malloc( 20 * sizeof(char) );
    if( mem_allocation== NULL ){
        printf("Couldn't able to allocate requested memory\n");
    }
    else{
        strcpy( mem_allocation,"Your name and surname");
    }
    printf("Dynamically allocated memory content : " \
        "%s\n", mem_allocation );
    free(mem_allocation);
}
```

3. Öncelikle sadece adınız için gerekli olan minimum bellek alanını malloc ile ayırınız. Daha sonra soyadınızı da eklemek için bu bellek alanını realloc ile genişletiniz.

First, allocate the minimum memory space required for your name with malloc. Then expand this memory area with realloc to include your last name.

Çözüm:

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
```

```
int main(){
    char *mem_allocation;
    /* memory is allocated dynamically */
    mem_allocation = (char*)malloc( 7 * sizeof(char) );
    if( mem_allocation == NULL ){
        printf("Couldn't able to allocate requested memory\n");
    }
    else{
        strcpy( mem_allocation,"Caner");
    }
    printf("Dynamically allocated memory content : " \
        "%s\n", mem_allocation );
    mem_allocation=(char*)realloc(mem_allocation,20*sizeof(char));
    if( mem_allocation == NULL ){
        printf("Couldn't able to allocate requested memory\n");
    }
    else{
        strcpy( mem_allocation,"Caner Ozcan");
    }
    printf("Resized memory : %s\n", mem_allocation );
    free(mem_allocation);
}
```

Ek Sorular

Soru 1: Aşağıdaki kod parçasının ekran çıktısı ne olmalıdır?

Question 1: What should be the screen output of code below?

```
#include <stdio.h>

void main()
{
    int c, *b, **a;

    c = 4;

    b = &c;

    a = &b;

    printf("%d", f(c, b, a));
}

int f(int x, int *py, int **ppz) {
    **ppz += 1;

    x = *py***ppz;

    *py *= 2;

    x *= *py+3;

    return (x+*py+**ppz);
}
```

Soru 2: Pointer kullanarak bir dizi tanımlayınız ve dizinin boyutunu kullanıcıya sorunuz. Dizinin elemanlarını kullanıcıdan alınız. Daha sonra dizide yeni bir alan oluşturun ve yeni elemanları kullanıcıdan alınız. En son yeni dizinin elemanlarını ekrana basınız. (Dinamik bellek yönetimi kullanılmalı.)

Question 2: First, define an array and ask the user for array size. Then, get the elements of array from user, too. After that, allocate new space in the array and get new elements. Finally, dump the resulting array to scree. (Use dynamic memory allocation.)

Soru 3: Bir cümleinin kelimelerini sondan başa doğru yazan programı dinamik bellek yönetimi kullanarak oluşturunuz.

Question 3: Write a program using dynamic memory allocation that prints out words of a sentence in reverse order.

Örnek (Example):

Input: I love Programming Languages course

Output: course Languages Programming love I

Converts a number given in base 10 to the base given by the user(between 2 and 9) and prints it on the screen write the code for the algorithm. The number in the new base must be stored in an array (note: to convert to a new base use % with the new base to get the result and / base for the remain, example to binary number%2 then number/2) .

Solution:

```
baseConversion2.c X
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  void main(){
5      int orgNumber= 123456789; // number to be converted
6      int i, base, k;
7      int result[100];
8      printf(" Please enter the new base between 2 and 9 to convert %d : ",orgNumber);
9      scanf("%d",&base); // read the needed base
10     i=0;
11     while (orgNumber > base-1) // continue till the orgNumber become less than the base
12     {
13         result[i]= orgNumber % base; // get the base
14         orgNumber = orgNumber / base; // the remain after get the base
15         i++; // counter for the number of elements
16     }
17     result[i]= orgNumber; // put the remain in the array
18
19     printf("\n Converted the number to %d base : ", base);
20
21     for (k=i;k>=0;k--)
22         printf("%d", result[k]); |
23
24
25     printf("\n");
26 }
27
```

Example 1

- ▶ Write a program that continuously takes a character unless user press ENTER and prints “*” for each character entered from keyboard.
- ▶ When user presses ENTER the program will write all the characters entered since the beginning of data entrance in input order. Character code for “ENTER” is 13.

Example 1

- ▶ Without dynamic memory allocating it will be limited.

```
#include<stdio.h>
#include<conio.h>

int main(){
    char giris[50];
    char *p;
    int i=0,k;
    p=giris;
    while(1){
        *(p+i)=getch();
        if(*(p+i)==13)
            break;
        putchar('*');
        printf("Adres[%d]: %d\n",i, p+i);
        i++;
    }
    printf("\n");
    for(k=0;k<i;k++){
        printf("Adres[%d]: %d\n",k, p+k);
        putchar(*(p+k));
    }
    getch();
    return 0;
}
```

Example 1

```
#include <stdio.h>
#include <stdlib.h>
void main()
{
    char *p;
    int i=0, k;

    p = (char *) malloc(sizeof(char));
    while(1)
    {
        *(p+i) = getch();
        if(*(p+i) == 13) break;
        putchar('*');
        i++;
        p = (char *) realloc(p, (i+1)*sizeof(char));
    }

    putchar('\n');
    for(k=0;k<i;k++)
        putchar(*(p+k));
}
```


Example 2

- Write a function with prototype given below which interchanges two variables values.

void swap (int*, int*)

```
#include <stdio.h>
void swap(int * q,int * p)
{
    int temp = *p;
    *p = *q;
    *q = temp;
}

int main()
{
    int a = 10, b = 2, x = 3, y = 5;
    printf("a,b,x,y: %d,%d,%d,%d\n", a, b, x, y);
    swap(&x, &y);
    swap(&a, &b);
    printf("a,b,x,y: %d,%d,%d,%d\n", a, b, x, y);
}
```

Example 3

- Write a function with prototype given below which calculates the area and perimeter of a rectangle.

void rectangle(int a,int b, int *area, int *perimeter)

```
#include <stdio.h>
void rectangle(int a, int b, int *area, int *perimeter);
int main() {
    int x, y;
    int area, perimeter;
    printf("Please enter the length and the width of the rectangle:\n " );
    scanf("%d %d", &x, &y);
    rectangle(x, y, &area, &perimeter);
    printf("the area= %d    , the perimeter = %d\n", area, perimeter);
}
void rectangle(int a,int b, int *area,int *perimeter) {
    *area = a * b;
    *perimeter = 2 * (a + b);
}
```

Example 4

- Write a function that performs like strlen function. Prototype for this function is as given below:

int StrLen(char *)

```
#include <stdio.h>
#include <conio.h>

int StrLen(char *);

int main() {
    char str[100];
    printf("Enter string : ");
    gets(str);
    printf("Length of string : %d\n", StrLen(str));
    getch();
}

int StrLen(char * p) {
    int n = 0;
    while(*p != '\0') {
        n++;
        p++;
    }
    return n;
}
```

PD2 LAB QUESTIONS

1. Öncelikle sadece adınız için gerekli olan minimum bellek alanını malloc ile ayırınız. Daha sonra soyadınızı da eklemek için bu bellek alanını realloc ile genişletiniz.

First, allocate the minimum memory space required for your name with malloc. Then expand this memory area with realloc to include your last name.

2. Klavyeden girilen iki kelimenin aynı olup olmadığını aynı değillerse hangisinin kelime olarak daha uzun olduğunu belirten program kodunu yazınız.

The program code that identifies whether two string entered from the keyboard are the same or not, and which one is longer as a word.

3. Verilen cümle içerisinde verilen kelimeyi arayan ve eğer kelime cümlede varsa bulunduğu yerlerin başlangıç indisini ekrana yazdıran programın kodunu yazınız.

Write a program that searches for the given string in the given sentence and if the word is found then prints the starting index of the places of this word to the screen.

4. Klavyeden girilen text bir ifadeyi okuyan ve bu text in içindeki harflerin sayısını veren programı dinamik bellek yönetimi kullanarak yazınız.

Örnek : “Karabük Üniversitesi” için;

2 adet “a”, 1 adet “b”....

Write a program that reads a string entered from the keyboard and find frequency of characters in this string using dynamic memory management.

Example: For “Karabük University”;

2 “a”, 1 “b”....

5. Belli bir metinde veya cümledeki kelimeleri sayan C programını dinamik bellek kullanarak yazınız.

Write the C program using dynamic memory, which counts the words in a given text or sentence.