

Câu 2. Kết quả khi thực thi các chương trình sau là gì?

```
#include <stdio.h>
int main() {
    int i = 1;
    int S = 0;
    int n = 36;
    while (i <= n) {
        if (n % i == 0) {
            S = S + (n % i);
        }
        i++;
    }
    printf("%d", S);
    return 0;
}
```

```
#include<stdio.h>
#include<math.h>
int main() {
    int n = 123456;
    int s = 0;
    while (n > 0) {
        s = (s * 10) + (n % 10);
        n = n / 10;
    }
    printf("%d", s);
    return 0;
}
```

```
#include <stdio.h>
int main() {
    int i = 0;
    printf("%d", i++);
    printf("%d", ++i);
    return 0;
}
```

```
#include<stdio.h>
int main() {
    int ch = 'A';
    printf("%d\t%c\n", ch, ch);
    printf("%d\t%c", ch + 32, ch+32);
    return 0;
}
```

Câu 3. Viết hàm đệ quy :

- a. Tính trung bình của mảng $a[5] = \{1, 2, 3, 4, 5\}$.
- b. Tìm số lớn nhất trong mảng $a[5] = \{1, 30, 20, 100, 96\}$.
- c. Tính giá trị biểu thức sau: $f(x) = \begin{cases} 1, & x = 0 \\ 2f(x-1) + \frac{3}{4}, & x > 0 \end{cases}$

Câu 4. Áp dụng cấu trúc Struct để giải quyết yêu cầu sau:

Một sinh viên có các thông tin bao gồm: Tên sinh viên, mã số sinh viên, năm sinh, điểm trung bình và số ngày nghỉ. Nhà trường cần nhập thông tin của tất cả các sinh viên đang theo học tại trường và lập ra danh sách các học sinh có thành tích tốt trong học tập (điểm trung bình lớn hơn 7.0) để khen thưởng và danh sách các học sinh có số ngày nghỉ lớn hơn 3 để ra nhắc nhở sinh viên đó.

Viết chương trình cho phép người dùng nhập số sinh viên cũng như thông tin của sinh viên, in thông tin của những sinh viên có thành tích tốt và những sinh viên bị nhắc nhở.

Lưu ý: có thể viết hàm và gọi trong hàm main().

Câu 5. Viết hàm kiểm tra một mảng chứa các kí tự có đối xứng hay không.

`int isSymmetrical(char input[], int length)`

Ví dụ: `abcba` \rightarrow 1,
 `abnmkgj` \rightarrow 0,
 `abcd dcba` \rightarrow 1.

Câu 2. Kết quả khi thực thi các chương trình sau là gì?

```
#include <stdio.h>
int main() {
    int i = 1;
    int S = 0;
    int n = 36;
    while (i <= n) {
        if (n % i == 0) {
            S = S + (n % i);
        }
        i++;
    }
    printf("%d", S);
    return 0;
}
```

0

```
#include<stdio.h>
#include<math.h>
int main() {
    int n = 123456;
    int s = 0;
    while (n > 0) {
        s = (s * 10) + (n % 10);
        n = n / 10;
    }
    printf("%d", s);
    return 0;
}
```

654321

```
#include <stdio.h>
int main() {
    int i = 0;
    printf("%d", i++);
    printf("%d", ++i);
    return 0;
}
```

02

```
#include<stdio.h>
int main() {
    int ch = 'A';
    printf("%d\t%c\n", ch, ch);
    printf("%d\t%c", ch + 32, ch+32);
    return 0;
}
```

65 A

97 a

Tính trung bình của mảng $a[5] = \{1, 2, 3, 4, 5\}$.

```
float recursiveAve(int a[], int n){  
    if(n == 1){  
        return (float)a[n-1];  
    }  
    else{  
        return ((float)(recursiveAve(a,n-1)*(n-1)+a[n-1])/n);  
    }  
}
```

Tìm số lớn nhất trong mảng $a[5] = \{1, 30, 20, 100, 96\}$.

TÌM GIÁ TRỊ LỚN NHẤT CỦA 2 SỐ

```
int max(int n1, int n2) {  
    return n1>n2?n1:n2;  
}
```

ĐỆ QUY

```
int recursiveMax(int a[], int length) {  
    if(length == 1) {  
        return a[0];  
    }  
    return max(recursiveMax(a, length-1), a[length-1]);  
}
```

Tính giá trị biểu thức sau: $f(x) = \begin{cases} 1, & x = 0 \\ 2f(x-1) + \frac{3}{4}, & x > 0 \end{cases}$

```
float bieuthuc(int x){  
    if(x == 0){  
        return 1;  
    }  
    else{  
        return 2.0*bieuthuc(x-1.0)+(3.0/4.0);  
    }  
}
```

...

Câu 4. Áp dụng cấu trúc Struct để giải quyết yêu cầu sau:

Một sinh viên có các thông tin bao gồm: Tên sinh viên, mã số sinh viên, năm sinh, điểm trung bình và số ngày nghỉ. Nhà trường cần nhập thông tin của tất cả các sinh viên đang theo học tại trường và lập ra danh sách các học sinh có thành tích tốt trong học tập (điểm trung bình lớn hơn 7.0) để khen thưởng và danh sách các học sinh có số ngày nghỉ lớn hơn 3 để ra nhắc nhở sinh viên đó.

Viết chương trình cho phép người dùng nhập số sinh viên cũng như thông tin của sinh viên, in thông tin của những sinh viên có thành tích tốt và những sinh viên bị nhắc nhở.

Lưu ý: có thể viết hàm và gọi trong hàm main().

```
typedef struct {  
    int day, month, year;  
} date_t;
```

```
typedef struct {  
    char name[100];  
    char mssv[8];  
    float dtb;  
    int ngaynghỉ;  
    date_t birthday;  
} sinhvien_t;
```

```

void nhapdulieuSV(sinhvien_t a[], int *n) {
    int i;
    for(i=0; i<n; i++) {
        printf("\n-----\n");
        fflush(stdin);
        printf("\nNhap Ho Va Ten SV: ");
        gets(a[i].name);
        fflush(stdin);
        printf("\nNhap MSSV: ");
        gets(a[i].mssv);
        fflush(stdin);
        printf("\nNhap ngay sinh: ");
        scanf("%d",&a[i].birthday.day);
        printf("\nNhap thang sinh: ");
        scanf("%d",&a[i].birthday.month);
        printf("\nNhap nam sinh: ");
        scanf("%d",&a[i].birthday.year);
        printf("\nNhap diem trung binh: ");
        scanf("%f",&a[i].dtb);
        printf("\nNhap so ngay nghi: ");
        scanf("%d",&a[i].ngaynghi);
    }
}

```

```

void xuatdulieuHS(sinhvien_t a[],int n) {
    int i;
    for(i=0; i<n; i++) {
        printf("\n-----\n");
        printf("\nHo va ten: %s",a[i].name);
        printf("\nMSSV: %s",a[i].mssv);
        printf("\nNgay sinh: %s/%s/%s",a[i].birthday.day,a[i].birthday.month,a[i].birthday.year);
        printf("\nDiem trung binh: %0.1f",a[i].dtb);
        printf("\nNgay nghi: %d",a[i].ngaynghi);
        printf("\n-----\n");
    }
}

```



```

void thanhtichtot(sinhvien_t a[],int n) {
    int i;
    for(i=0; i<n; i++) {
        if(a[i].dtb > 7.0) {
            printf("\n-----\n");
            printf("\nHo va ten: %s",a[i].name);
            printf("\nMSSV: %s",a[i].mssv);
        }
        else{
            printf("Khong co sinh vien co dtb > 7");
        }
    }
}

```

```

void nghiphepnhieu(sinhvien_t a[],int n) {
    int i;
    for(i=0; i<n; i++) {
        if(a[i].ngayngghi > 3) {
            printf("\n-----\n");
            printf("\nHo va ten: %s",a[i].name);
            printf("\nMSSV: %s",a[i].mssv);
        }
        else{
            printf("Khong co sinh vien co dtb > 7");
        }
    }
}

```

Câu 5. Viết hàm kiểm tra một mảng chứa các kí tự có đối xứng hay không.

```
int isSymmetrical(char input[], int length)
```

Ví dụ: abcba → 1,

 abnmkgj → 0,

 abcd dcba → 1.

```
int ktdoixung(char s[]){
    int i;
    for(i=0;i<strlen(s)/2;i++){
        if(s[i] != s[strlen(s)-1-i]){
            return 0;
        }
        else{
            return 1;
        }
    }
}
```