

ELEKTRIJADA 2023

Ohrid (North Macedonia), May 20-24, 2023

INFORMATICS

Please, suppose that Little Endian is used in all tasks, and IEEE-754 is used for FP data, and
sizeof(char)=1 sizeof(short)=2 sizeof(int)=sizeof(int*)=sizeof(long)=sizeof(float)=4
sizeof(long long)=sizeof(double)=8

1 Find the output of the following program:

```
#include <stdio.h>
#define X __func__
#define P(X) printf("%s", X)

unsigned char z='1', fct='1';
void f() { P(X); z++; }

int main()
{
    void f();
    for (z=fct='1'; z<'4'; z++)<%
        fct *= z;
    printf("%c: sqr=%d f=%d\n", z, z*z, fct*=z);%>
    printf(__func__);
    return 0;
}
```

2 Find the output of the following program:

```
#include <stdio.h>
#define F(x) printf("%s", __func__)
#define M 2

typedef void (*pf[2])();
typedef pf m[][M];

static int t;

void f1() { if(~t) t++; F(f1); }
void f2() { F(f1); }

void f3()
{
    F(f3);
    printf(":%x", (unsigned int)sizeof(pf));
}

void f4()
{
    F(f4);
    if(!076&1|0x30>>2<<4) f1(); t++;
}

int main()
{
    m m={f1,f2,f3,f4,f1,f2,f3,f4};
    for(int i = 0; i < M; i++)
        for(int j = 0; j < M; j++,printf("\n"))
            if(!i) m[i][j][0]();
            else (*(m+i*M-1))[i]();
    printf("t:%d", t);
    return 0;
}
```

3 Find the output of the following program:

```
#include <stdio.h>
#define P(x) printf("%d",x)
typedef struct s
{ int n, m[10][10][2]; } S;

void f0(S *s)
{
    int i, j, k, z;
    for(k=1; k<2; k++)
        for (i=0; i<s->n; i++)
            for (j=0; j<s->n; j++)
                if (s->m[i][j][k-1]&& s->m[i][j][k-1]<19)
                    s->m[i][j][k]=i+1;

    void f1(int i, int j, int k)
    {
        if (i==j) P(i);
        else
        {
            f1(i, s->m[i-1][j-1][!!k], k);
            P(j);
        }
    }

    for (z=0; z<1; z++)
        for (k=0; k<s->n; k++)
            for (i=0; i<s->n; i++)
                for (j=0; j<s->n; j++)
                    if (s->m[i][j][z]>
                        s->m[i][k][z]+s->m[k][j][z])
                    {
                        s->m[i][j][z]=
                            s->m[i][k][z]+s->m[k][j][z];
                        s->m[i][j][z+1]=s->m[k][j][z+1];
                    }

    for (i=s->n; i; i--)
        for (j=i-1; j; j--)
            f1(i, j, j), printf(":%d\n",
                                s->m[i-1][j-1][!i&!j]);
}

int main()
{
    S s={4, {
        {{0},{2},{8},{20}},
        {{20},{0},{3},{4}},
        {{20},{20},{0},{7}},
        {{5},{20},{20}}
    }};
    f0(&s);
    return 0;
}
```

4 Find the output of the following program:

```
#include <stdio.h>
int main()
{
    union U {char c[4]; float f[4]; int i[4];}
    u={{[0]=24.5f}};
    printf("1:%x\n", u.i[0]);
    printf("2:%x\n", u.i[1]);
    u.c[2]=(u.c[1]=(u.c[0]=010>>01)>>01)>>01;
    printf("3:%x\n", u.i[0]);
    printf("4:%.2f\n", u.f[0]);
    return 0;
}
```

5 Find the output of the following program in case of its successful execution:

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

char* zz(char* s, int n)
{
    int key=-1;
    while (s[++key]);
    printf("%d-",key);

    char *r;
    r=(char *)calloc(key+1, sizeof(char));
    if (n<=1) { while(*r++=*s++); }
    else
    {
        for (int rn=0; rn<n; rn++)
        {
            int i=rn;
            int u=1;
            while (i<key)
            {
                *r++=(s+i);
                if (rn==0||rn==n-1)
                { i+=(2*n-2); }
                else
                {
                    if (u) { i+=(2*(n-rn)-2); }
                    else { i+=rn*2; }
                    u^=1;
                }
            }
        }
    }
    return r-key;
}

int main()
{
    char *ss,
    *s[] = { "OHRIDNM", "INFORMATICS23" };
    for (int i=0; i<sizeof(s)/sizeof(*s); i++)
        printf("%s\n", (ss=zz(s[i],i+2))),
        free(ss);
    return 0;
}
```

6 Find the output of the following program:

```
#include <stdio.h>
#define T(x,y) (x<y)?x:y
static int key;
int v[]={2,3,1,1,4,1};
int f1()
{
    if(!key)
    {
        int z=0, x=1;
        do key=z,z|=x,x<=1; while(x);
        printf("f1: %d %d %d\n", key<<1, z, x);
    }
    return key;
}

int f0(int n)
{
    int m[100]; int i, j;
    if (n == 0 || v[0] == 0) return f1();
    m[0] = 0;

    for (i=1; i<n; i++)
    {
        m[i]=f1();
        for (j=0; j<i; j++)
            if (i<=j+v[j]&& m[j]!=f1())
            {
                m[i] = T(m[i],m[j]+1);
                break;
            }
        printf("%d", m[i]);
    }
    return m[n-1];
}

int main()
{
    int i;
    for (i=3; i--; )
        printf(":%d\n",
            f0(sizeof(v)/sizeof(v[i])-i));
    return 0;
}
```

7 Find the output of the following program:

```
#include <stdio.h>

unsigned long int cat(unsigned int n)
{
    if (n<=1) return 1;
    unsigned long int res=0;
    for (unsigned i=0; i<n; i++)
        res += cat(i)*cat(n-i-1);
    return res;
}

int main()
{
    for(int i=0; i<10; i+=5)
        for (int j=1;j<5;j++)
            printf("dog:%u\n",cat(i+++j++));
    return 0;
}
```

8 Find the output of the following program in case of its successful execution:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define O 1<<1
#define H(A) (A)*(A)
#define R(A,B) A=(B*)calloc(1,sizeof(B))
#define I(A) memcpy(A->a,a,sizeof(*a)*(O))
#define D(A,B) R(A,B),I(A)
#define NM(A,B) printf("%d-%d\n", A, B);

typedef struct n
{ int a[O]; struct n *n[O]; } n;
static int m, v, d=-1;

n* f1(n *c, int a[O])
{
    int y=m++;
    if (!c) D(c,n);
    else c->n[a[y%2]>c->a[y%2]]=
        f1(c->n[a[y%2]>c->a[y%2]],a);
    return c;
}

int f2(n n1, n n2)
{
    int i=0, d=0;
    while (d--) i+=H(n1.a[d]-n2.a[d]);
    return i;
}

void f3(n *c, int a[O])
{
    if (!c) return; v++;

    if (d== -1 || f2(*c,(n){a[0],a[1]})<d)
        d=f2(*c,(n){a[0],a[1]});

    if (!d) return;
    int y=m++; m%=2;
    f3(c->n[a[y]>c->a[y]],a);
    if (H(c->a[y]-a[y])>=d) return;
    f3(c->n[a[y]<c->a[y]],a);
}

void f4(n *c)
{
    if (!c) return;
    f4(c->n[m%2]);
    f4(c->n[(m+1)%2]);
    NM(c->a[m%2],c->a[(m+1)%2]);
    free(c);
}

int main()
{
    n *c=0;
    int i, p[][2]=
        {7,2,5,4,2,3,4,7,9,6,8,1,6,6};
    for (i=0; i<6; i++) c=f1(c,p[i]), m=0;
    f3(c,p[i]),printf("d:%d v:%d\n",d, v);
    f4(c);
    return 0;
}
```

Points/Task Distribution

1	2	3	4	5	6	7	8	Σ
10	10	18	10	14	12	8	18	100

Appendix: ASCII table

	0	1	2	3	4	5	6	7
0	NUL	DLE	space	0	@	P	`	p
1	SOH	DC1 XON	!	1	A	Q	a	q
2	STX	DC2	"	2	B	R	b	r
3	ETX	DC3 XOFF	#	3	C	S	c	s
4	EOT	DC4	\$	4	D	T	d	t
5	ENQ	NAK	%	5	E	U	e	u
6	ACK	SYN	&	6	F	V	f	v
7	BEL	ETB	'	7	G	W	g	w
8	BS	CAN	(8	H	X	h	x
9	HT	EM)	9	I	Y	i	y
A	LF	SUB	*	:	J	Z	j	z
B	VT	ESC	+	;	K	[k	{
C	FF	FS	,	<	L	\	l	
D	CR	GS	-	=	M]	m	}
E	SO	RS	.	>	N	^	n	~
F	SI	US	/	?	O	_	o	del