

ELEKTRIJADA 2019

Sunny Beach (Bulgaria), May 18-23, 2019

INFORMATICS

Please, suppose in all tasks that Little Endian is used, and
`sizeof(int)=sizeof(int*)=sizeof(float)=4` `sizeof(long long)=sizeof(double)=8`

1 Find the output of the following program:

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

typedef struct x {char c; struct x *p;} X;
X *f2() { return (X*) calloc(1,sizeof(X)); }

void f1(X **p, char c)
{
    X *t, *n=f2(); (*n).c=c;
    if (!*p) *p=n; else
    { for (t=*p; (*t).p; t=(*t).p); (*t).p=n; }
}

void f3(const X* p)
{
    for (; (*p).p; p=p->p) printf("%c", p->c);
}

unsigned int f6(float c)
{
    unsigned int* u=(unsigned int*) &c, i=0;
    for (;;) if (*u<1<i++) return i;
}

int fc(char a, char b, unsigned (*f)(float))
{
    if (!f) return a<b; return (*f)(a)<(*f)(b);
}

X *f4(X **p, unsigned (*f)(float))
{
    X* t=(*p).p,*t1,*t2;
    for ( ; t->p; t=t->p)
    {
        char c=t->c;
        for (t2=*p; t2->p!=t; t2=t2->p);
        for (t1=t; t1!=*p && fc(c,t2->c,f));
        {
            t1->c=t2->c; t1=t2;
            for (t2=*p; t2->p!=t1&&t1!=*p; t2=t2->p);
        }
        t1->c=c;
    }
    return *p;
}

int main()
{
    const char* c="Beach";
    X *p1;
    for (p1=0; *c; f1(&p1,*c++));
    printf("01: "),f3(p1),printf("\n");
    printf("02: "),f3(f4(&p1,0)),printf("\n");
    printf("03: "),f3(f4(&p1,f6)),printf("\n");
    return 0;
}
```

2 Find the output of the following program:

```
#include <stdio.h>
#define P(s) printf("%c", *s)
void f0(char *s, int i, char *c)
{
    if (!*(s+i-1)) *(s+i-1)=*c;
    else if (*c<*(s+i-1)&&2*i<32) f0(s,2*i,c);
    else if (*c>*(s+i-1)&&2*i+1<32)
        f0(s,2*i+1,c);
}

void f1(char *s, int i, int t)
{
    int j=0;
    for (; j<32; j++, s++) if (*s++) P(--s);
}

void f(char *s, int i, int t)
{
    if (i<32 && *(s+i-1) && t==2)
        P(s+i-1), f(s,2*i,t), f(s,2*i+1,t);
    else if (i<32 && *(s+i-1) && t==3)
        f(s,2*i,t), P(s+i-1), f(s,2*i+1,t);
    else if (i<32 && *(s+i-1))
        f(s,2*i,t), f(s,2*i+1,t), P(s+i-1);
}

int main()
{
    int i;
    char *c="SUNNY", s[100]={};
    void (*g[])(char*,int,int) = {f1,f};
    void (**h)(char*,int,int) = g;
    while (*c) f0(s,sizeof(*c),c++);
    for (i=1; i<5; h+=i+=1)
        printf("%d:",i),(**h)(s,1,i),printf("\n");
    return 0;
}
```

3 Find the output of the following program:

```
#include <stdio.h>
typedef struct fi { int d, n; } FI;
int main(void)
{
    int i, k, n, br;
    for (i=1; i<4; i++)
    {
        FI fi1={0,1}, fi2={1,3*i}, t;
        for (n=3*i, br=2; fi2.n>1; br++)
        {
            k=(n+fi1.n)/fi2.n, t=fi1, fi1=fi2;
            fi2=(FI){fi2.d*k-t.d,fi2.n*k-t.n};
        }
        printf("%d:%d\n", i, br);
    }
    return 0;
}
```

4 Find the output of the following program:

```
#include <stdio.h>
struct F { unsigned char f, t, d, s; };
struct F x[] = {{ 'N', 'C', 300 }, { 'C', 'D', 300 },
               { 'N', 'T', 200 }, { 'N', 'D', 400 }, { 'T', 'L', 400 },
               { 'D', 'H', 300 }, { 'H', 'L', 400 }, { 'D', 'L', 500 } };
struct F ss[20], btd[20];
int fp, y, ts, gd, gt, gs;
void fr(char tt)
{
    for (gd=gt=0; gt<ts; gt++)
        printf("%c-", ss[gt].f), gd+=ss[gt].d;
    printf("%c:%d\n", tt, gd);
}
void cm() { for(gt=0; gt<fp; x[gt++].s=0); }
void ret(char f, char t)
{
    for(gt=0; gt<fp; gt++)
        if(x[gt].f==f&&x[gt].t==t)
            { x[gt].f=0; return; }
}
int fm(char a, char b)
{
    for(gt=fp-1; gt>-1; gt--)
        if(x[gt].f==a&&x[gt].t==b) return x[gt].d;
    return 0;
}
int fn(char f, char *a)
{
    for(y=0; y<fp; y++)
        if (x[y].f==f&&!x[y].s)
            { *a=x[y].t, x[y].s=1; return x[y].d; }
    return 0;
}
void push(char f, char t, int d)
{
    if (ts<20)
        ss[ts].f=f, ss[ts].t=t, ss[ts++].d=d;
}
void pop(char *f, char *t, int *d)
{
    if (ts>0)
        *f=ss[--ts].f, *t=ss[ts].t, *d=ss[ts].d;
}
void isf(char f, char t)
{
    char a;
    if (gd=fm(f,t)) { push(f,t,gd); return; }
    if (gd=fn(f,&a)) push(f,t,gd), isf(a,t);
    else if (ts>0) pop(&f,&t,&gd), isf(f,t);
}
int main(void)
{
    char f='N', t='L', c, d;
    fp = sizeof(x)/sizeof(struct F);
    for (gd=gs=0; gd<fp; gs+=x[gd++].d);
    printf("01:%d\n",gs);
    for (gs=2; gs<5; gs++, ts=0)
    {
        printf("%02d:",gs);
        isf(f,t),fr(t),cm();
        if (ts>0) pop(&c,&d,&gd); ret(c,d);
    }
    return 0;
}
```

5 Find the output of the following program:

```
#include <stdio.h>
#define MAX 3
struct X { unsigned int d[MAX], f, r, rc; };
int fp(struct X *x, int d)
{
    if (x->f==x->r) x->rc++;
    x->d[x->r]=x->d[x->r]<<8|d;
    x->r=(x->r+1)%MAX;
    return 1;
}
int fg(struct X *x, int *d)
{
    if (!x->rc) return 0;
    x->rc=x->rc>4?4:x->rc;
    *d=x->d[x->f]>>8*(x->rc%5-1);
    x->d[x->f]<=<=32-8*(x->rc%5-1);
    x->d[x->f]>>=32-8*(x->rc%5-1);
    x->f=(x->f+1)%MAX;
    if ((x->r+1)%MAX==x->f) x->rc--;
    return 1;
}
int main()
{
    struct X x={}; int i,j,c;
    for (i=1; i<15; i++) fp(&x,i);
    printf("01:%08X\n", x.d[x.r]);
    for (i=2; i<5; i++)
    {
        printf("%02d:", i);
        for (j=1; j<4; j++)
            if (fg(&x,&c)) printf("%X", c);
        printf("\n");
    }
    return 0;
}
```

6 Find the output of the following program:

```
#include <stdio.h>
#include <string.h>
#define pstr(s) printf("-%d\n",printf(s))
#define qstr(s,y) printf("-%d\n",printf(s,y))
#define ystr(s) str(s##s)
#define xstr(s) str(s)
#define str(s) #s
#define foo 54
int main()
{
    const char *p="SUNNY BEACH";
    const char *pq="INF";
    int i;
    i = pstr("01:"xstr(pq));
    i = pstr("02:"xstr(i));
    i = pstr("03:"str(p="BEACH\n");));
    i = qstr("04:%s",p);
    i = qstr("05:%d",i);
    i = qstr("06:"xstr(*pq),i);
    i = printf("07:.*s\n",i,p);
    i = qstr("08:%s",ystr(foo));
    i = qstr("09:%s",str(*pq));
    i = qstr("%s",printf("10:")p);
    return 0;
}
```

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

```
#include <stdio.h>
struct X { short int s; char c; };
const char *fw()
{
    struct X x[] = {
        {0x414243, 'a'}, {0x444546, 'b'},
        {0x313233, 'c'}, {0x343536, 'd'}};
    char *s="FILE.DAT";
    int i;
    FILE *fp=fopen(s,"wb+");
    if (fp)
    {
        for(i=0;i<sizeof(x)/sizeof(x[i]); i++)
        {
            fwrite(&x[i].s,sizeof(x[i].s),1,fp);
            fwrite(&x[i].c,sizeof(x[i].c),1,fp);
        }
        fclose(fp);
    }
    return s;
}
void fr(const char *s)
{
    struct X x;
    int i;
    FILE *fp=fopen(s,"rb+");
    if (fp)
    {
        for (i=0; fread(&x,sizeof(x),1,fp); )
            printf("%02d:%x %c\n", ++i, x.s, x.c);
        fclose(fp);
    }
}
int main()
{
    fr(fw());
    return 0;
}
```

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

Points/Task Distribution

1	2	3	4	5	6	7	Σ
15	18	13	20	13	10	11	100