

GRUPA A  
Zawartość pliku defs.h

```
#include <stdio.h>

#define PR(format, val) printf\
    (#val " = %" #format "\t", (val))
#define NL putchar('\n')

#define PRINT1(f,x1) PR(f,x1), NL
#define PRINT2(f,x1,x2) PR(f,x1), PRINT1(f,x2)
#define PRINT3(f,x1,x2,x3) PR(f,x1), \
    PRINT2(f,x2,x3)
```

Zad. 1. (5pkt.)

Podaj rezultat działania programu.

```
\begin{verbatim}
#include <stdio.h>
char *c[] = {
    "HOOD",
    "LATE",
    "RING",
    "BABY"
};
char **cp[] = {c+3, c+2, c+1, c};
char ***cpp=c;
```

```
void main(void)
{
    printf("%s",***+cpp + 3);
    printf("%s", *--***+cpp + 1);
    printf("%s", *cp[-2] + 2);
    printf("%s", cpp[-1][-1] + 3);
}
```

Zad. 2. (5 pkt.) Podaj rezultat działania programu.

```
#include "defs.h"

int next (void);
int last (void);
int new (int);
int reset (void);

int i=4;

void main (void)
{
    auto int i, j;

    i = reset();
    for (j=4; j<=6; j++) {
        PRINT2(d,i,j);
        PRINT1(d,next());
        PRINT1(d,last());
        PRINT1(d,new(i+j));
    }
}
```

W jednym pliku:

```
static int i=8;

int next (void)
{
    return ( i+=3 );
}
```

```
}

int last (void)
{
    return ( i--1 );
}

int new (int i)
{
    static int j=5;
    return ( i=j+=i );
}
```

W drugim pliku:

```
extern int i;

int reset (void)
{
    return (--i);
}
```

Zad. 3. (5pkt.) Podaj rezultat działania programu.

```
#include <stdio.h>
void main()
{ int n=10, p=5, q=10, r;
  r=n==(p=q);
  printf("A: n=%d p=%d q=%d r=%d\n",n,p,q,r);
  n=p=q=r=5;
  n+=p+=q;
  printf("B: n=%d p=%d q=%d\n",n,p,q);
  q=n<p?n++:p++;
  printf("C: n=%d p=%d q=%d\n",n,p,q);
  q=n>p?++n:++p;
  printf("D: n=%d p=%d q=%d\n",n,p,q);

  n=5; p=2; /* przypadek 5 */
  q=n++ > p || p++ != 3;
  printf("E: n=%d p=%d q=%d\n",n,p,q);
  n=5; p=2; /* przypadek 6 */
  q=n++ < p || p++ != 3;
  printf("F: n=%d p=%d q=%d\n",n,p,q);
  n=5; p=2; /* przypadek 7 */
  q=++n == 3 && ++p == 3;
  printf("G: n=%d p=%d q=%d\n",n,p,q);
  n=5; p=2; /* przypadek 8 */
  q=++n == 6 && ++p == 3;
  printf("H: n=%d p=%d q=%d\n",n,p,q);
}
```

Zad. 4. (5pkt.) Podaj rezultat działania programu.

```
#include <stdio.h>
char input []="SUPEKWAPEDERAT1\1\11W\1PIECZEBYE";

void main(void)
{
    int i,c;
    for (i=2; (c=input[i]) != '\0'; i++) {
        switch (c) {
            case 'a': putchar('i'); continue;
            case '1': break;
            case 1: while ( (c=input[++i]) != '\1'
                && c != '\0');
            case 9: putchar ('L');
            case 'E': case 'P': continue;
            default: putchar (c); continue;
        }
    }
}
```

```

    }
    putchar ( ' ' );
}
putchar ('\n');
}

```

Zad. 5. (5pkt.) Podaj rezultat działania programu.

```

#include "defs.h"

int reset(void);
int next(int);
int last(int);
int new(int);

int i=3;

void main(void)
{
    auto int i,j;

    i=reset();
    for (j=3; j<=5; j++) {
        PRINT2(d,i,j);
        PRINT1(d,next(i));
        PRINT1(d,last(i));
        PRINT1(d,new(i+j));
    }
}

int reset(void)
{
    return i--;
}

int next(int j)
{
    return (j-=++i);
}

int last(int j)
{
    static int i=10;
    return (i+=--j);
}

int new(int k)
{
    auto int j=10;
    return (i=j-=k);
}

```

Zad. 6. (5pkt.) Napisać definicję makra *cubabs(a)*, które wyznaczy trzecią potęgę wartości bezwzględnej swojego argumentu ( $|a|^3$ ).

Zad. 7. (5pkt.) Napisać krótki program z argumentami, który będzie przyjmował jako argumenty "+a" albo "+b" i drukował na ekranie komunikaty zgodnie z poniższą tabelą:

Wydruk	Param.
"Prawidłowe wywołanie!! Tryb pracy 1!!"	+a
"Prawidłowe wywołanie!! Tryb pracy 2!!"	+b
"Zły parametr!! Nie wykonuje akcji!! "	inne

Zad. 8. (5pkt.) Dana jest funkcja *reverse(s)* odwracająca znaki tekstu *s*. Napisać jej rekurencyjną wersję.

```

void reverse(char s[])
/* odwroc tekst w miejscu */
{
    int c,i,j;
    for (i=0, j=strlen(s)-1; i<j; i++,j--) {
        c=s[i];
        s[i]=s[j];
        s[j]=c;
    }
}

```

Zad. 9. (5 pkt.) Podaj rezultat działania programu.

```

#include <stdio.h>
#include "defs.h"

struct s1 {
    char *s;
    int i;
    struct s1 *s1p;
};

void main(void)
{
    static struct s1 a[] = {
        {"KASA",1,a+1},
        {"LADA",2,a+2},
        {"CENA",3,a}
    };
    struct s1 *p=a;
    int i;
    PRINT3(s,a[0].s,p->s,a[2].s1p->s);

    for (i=0; i<2; i++) {
        PR(d,--a[i].i);
        PR(c,++a[i].s[3]);
        NL;
    }

    PRINT3(s, ++(p->s), a[(++p)->i].s,
        a[--(p->s1p->i)].s);
}

```