

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

#define FREEZING_POINT 32.0
#define SCALE_FACTOR (5.0 / 9.0)

int main(void) {
    setlocale(LC_ALL, "RU");
    double fahrenheit = 0.0;

    printf("Температура по шкале Фаренгейта: ");
    scanf("%lf", &fahrenheit);

    double celsius = (fahrenheit - FREEZING_POINT) * SCALE_FACTOR;

    printf("Температура по шкале Цельсия = %g\n", celsius);
    return EXIT_SUCCESS;
}
```

Содержимое файла *stdio.h*

Содержимое файла *stdlib.h*

Содержимое файла *locale.h*

Пустая строка

Пустая строка

Пустая строка

Пустая строка

```
int main(void) {  
    setlocale(LC_ALL, "RU");  
    double fahrenheit = 0.0;  
  
    printf("Температура по шкале Фаренгейта: ");  
    scanf("%lf", &fahrenheit);  
  
    double celsius = (fahrenheit - 32.0) * (5.0 / 9.0);  
  
    printf("Температура по шкале Цельсия = %g\n", celsius);  
    return 0;  
}
```

```
#define STR_LEN 80
#define TRUE      true
#define FALSE     0
#define GR        1.61803
#define CR        '\r'
#define EOS       '\0'
#define MEM_ERR   "Ошибка: недостаточно памяти"
                    ---

#define BEGIN {
#define END      }
#define LOOP for (;;)
#define BOOL int
#define DEBUG
```

```

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

#define MAX(x, y)      ((x) > (y) ? (x) : (y))
#define IS_EVEN(n)     ((n) % 2 == 0)
#define PRINT_INT(n)   printf("%d\n", n)

int main(void) {
    setlocale(LC_ALL, "RU");
    int i = 0, j = 3, k = 4,
        m = 5, n = 6;

    i = MAX(j + k, m - n);

    if (IS_EVEN(i)) {
        i++;
    }
    PRINT_INT(i);
    return EXIT_SUCCESS;
}

                                     ---
#define TOUPPER(c)      ('a' <= (c) && (c) <= 'z' ? \
                        (c) - 'a' + 'A' : (c))
#define getchr()       getc(stdin)
...
    getchr();
    printf("%c\n", TOUPPER('b'));

                                     ---
#define PRINT_INT(n)   printf(#n " = %d\n", n)
                                     ---

#define GENERIC_MAX(type) \
type##_max(type x, type y) { \
    return x > y ? x : y; \
}
GENERIC_MAX(double)

```

```

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

#define CHECK_ZERO(divisor) \
if (divisor == 0){ \
    printf("*** Попытка деления на ноль " \
        "в строке %d файла %s. ***\n\n", \
        __LINE__ + 1, __FILE__); \
    return EXIT_FAILURE; \
}

int main(void) {
    setlocale(LC_ALL, "RU");

    int i = 2, j = 0;

    printf("Программа скомпилирована %s в %s.\n\n", __DATE__, __TIME__);

    CHECK_ZERO(j);
    int k = i / j;

    return EXIT_SUCCESS;
}

```

```

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

#define TEST(condition, ...) ((condition) ? \
    printf("Тест %s пройден\n\n", #condition) : \
    printf(__VA_ARGS__))

```

```

int main(void) {
    setlocale(LC_ALL, "RU");
    int voltage = 15;
    int maxVoltage = 5;
    TEST(voltage <= maxVoltage,
        "Напряжение %d превосходит %d\n\n",
        voltage, maxVoltage);
    return EXIT_SUCCESS;
}

```

```

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

#define FUNCTION_CALLED() \
    printf("Вызвана функция %s\n\n", __func__);
#define FUNCTION_RETURNS() \
    printf("Возврат из функции %s\n\n", __func__);

void myFunction(void) {
    FUNCTION_CALLED();
    FUNCTION_RETURNS();
}

int main(void) {
    setlocale(LC_ALL, "RU");

    myFunction();

    return EXIT_SUCCESS;
}

```

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

```
#define DEBUG 1
```

```
int main(void) {
    setlocale(LC_ALL, "RU");

    int i = 5, j = 6, k = 0;
```

```
    #if DEBUG
        printf("Значение i: %d\n", i);
        printf("Значение j: %d\n", j);
    #endif
```

```
    return EXIT_SUCCESS;
```

```
}
```

```
    #if defined DEBUG // #ifdef, #ifndef
```

```
#define DEBUG    1
#define RELEASE  0
```

```
...
    #if DEBUG
        printf("Значение i: %d\n", i);
    #elif RELEASE
        printf("Значение j: %d\n", j);
    #else
        printf("Значение k: %d\n", k);
    #endif
```
