

数据类型

DATA TYPES AND EXPRESSIONS
2 数据类型与表达式



• 例：

整型变量

$-2^{31} \sim 2^{31}-1$

```
/*An Example*/  
#include <stdio.h>  
int main(void)  
{  
    int ix=3,iy=5,iz;  
    float fx=2.0,fy=5.5,fz;  
    char cx= 'A' ,cy= 'D' ,cz;  
    iz=ix+iy; fz=fx+fy; cz=cx+cy;  
    printf( "%d+%d=%d\n" ,ix,iy,iz);  
    printf( "%.2f+%.2f=%.2f\n" ,fx,fy,fz);  
    printf( "%c+%c=%c\n" ,cx,cy,cz);  
    return 0;  
}
```



• 例：

整数运算

ix	3
iy	5
iz	8

```
/*An Example*/  
#include <stdio.h>  
int main(void)  
{  
    int ix=3, iy=5, iz;  
    float fx=2.0, fy=5.5, fz;  
    char cx= 'A' ,cy= 'D' ,cz;  
    iz=ix+iy; fz=fx+fy; cz=cx+cy;  
    printf( "%d+%d=%d\n" ,ix,iy,iz);  
    printf( "%.2f+%.2f=%.2f\n" ,fx,fy,fz);  
    printf( "%c+%c=%c\n" ,cx,cy,cz);  
    return 0;  
}
```



• 例：

```
/*An Example*/  
#include <stdio.h>  
int main(void)  
{  
    int ix=3,iy=5,iz;  
    float fx=2.0,fy=5.5,fz;  
    char cx= 'A' ,cy= 'D' ,cz;  
    iz=ix+iy; fz=fx+fy; cz=cx+cy;  
    printf( "%d+%d=%d\n" ,ix,iy,iz);  
    printf( "%.2f+%.2f=%.2f\n" ,fx,fy,fz);  
    printf( "%c+%c=%c\n" ,cx,cy,cz);  
    return 0;  
}
```

输出计算结果

输出： 3+5=8



• 例：

实数变量



```
/*An Example*/
#include <stdio.h>
int main(void)
{
    int ix=3,iy=5,iz;
    float fx=2.0,fy=5.5,fz;
    char cx= 'A' ,cy= 'D' ,cz;
    iz=ix+iy; fz=fx+fy; cz=cx+cy;
    printf( "%d+%d=%d\n" ,ix,iy,iz);
    printf( "%.2f+%.2f=%.2f\n" ,fx,fy,fz);
    printf( "%c+%c=%c\n" ,cx,cy,cz);
    return 0;
}
```

fx

2.0

fy

5.5

fz

• 例：

实数变量

```
/*An Example*/
#include <stdio.h>
int main(void)
```

```
{
    int ix=3,iy=5,iz;
    float fx=2.0,fy=5.5,fz;
```

$-3.4 \times 10^{38} \sim 3.4 \times 10^{38}$

```
printf( "%.2f+%.2f=%.2f\n" ,fx,fy,fz);
printf( "%c+%c=%c\n" ,cx,cy,cz);
return 0;
}
```

fx	2.0
fy	5.5
fz	



• 例：

实数运算

fx	2.0
fy	5.5
fz	7.5

```
/*An Example*/
#include <stdio.h>
int main(void)
{
    int ix=3,iy=5,iz;
    float fx=2.0,fy=5.5,fz;
    char cx= 'A' ,cy= 'D' ,cz;
    iz=ix+iy; fz=fx+fy; cz=cx+cy;
    printf( "%d+%d=%d\n" ,ix,iy,iz);
    printf( "%.2f+%.2f=%.2f\n" ,fx,fy,fz);
    printf( "%c+%c=%c\n" ,cx,cy,cz);
    return 0;
}
```




• 例：

```
/*An Example*/  
#include <stdio.h>  
int main(void)  
{  
    int ix=3,iy=5,iz;  
    float fx=2.0,fy=5.5,fz;  
    char cx= 'A' ,cy= 'D' ,cz;  
    iz=ix+iy; fz=fx+fy; cz=cx+cy;  
    printf( "%d+%d=%d\n" ,ix,iy,iz);  
    printf( "%.2f+%.2f=%.2f\n" ,fx,fy,fz);  
    printf( "%c+%c=%c\n" ,cx,cy,cz);  
    return 0;  
}
```

输出实数
计算结果



输出： 2.00+5.50=7.50



• 例：

字符变量

-256~255

```
/*An Example*/  
#include <stdio.h>  
int main(void)  
{  
    int ix=3,iy=5,iz;  
    float fx=2.0,fy=5.5,fz;  
    char cx= 'A' ,cy= 'D' ,cz;  
    iz=ix+iy; fz=fx+fy; cz=cx+cy;  
    printf( "%d+%d=%d\n" ,ix,iy,iz);  
    printf( "%.2f+%.2f=%.2f\n" ,fx,fy,fz);  
    printf( "%c+%c=%c\n" ,cx,cy,cz);  
    return 0;  
}
```



• 例：

```
/*An Example*/  
#include <stdio.h>  
int main(void)  
{  
    int ix=3,iy=5,iz;  
    float fx=2.0,fy=5.5,fz;  
    char cx= 'A' ,cy= 'D' ,cz;  
    iz=ix+iy; fz=fx+fy; cz=cx+cy;  
    printf( "%d+%d=%d\n" ,ix,iy,iz);  
    printf( "%.2f+%.2f=%.2f\n" ,fx,fy,fz);  
    printf( "%c+%c=%c\n" ,cx,cy,cz);  
    return 0;  
}
```

字符运算



• 例：

3+5=8
2.00+5.50=7.50
A+D=?

输出实数
计算结果

```
/*An Example*/  
#include <stdio.h>  
int main(void)  
{  
    int ix=3,iy=5,iz;  
    float fx=2.0,fy=5.5,fz;  
    char cx= 'A' ,cy= 'D' ,cz;  
    iz=ix+iy; fz=fx+fy; cz=cx+cy;  
    printf( "%d+%d=%d\n" ,ix,iy,iz);  
    printf( "%.2f+%.2f=%.2f\n" ,fx,fy,fz);  
    printf( "%c+%c=%c\n" ,cx,cy,cz);  
    return 0;  
}
```

输出：A+D=?



• 例：

变量先定义

```
/*An Example*/  
#include <stdio.h>  
int main(void)  
{  
    int ix=3,iy=5,iz;  
    float fx=2.0,fy=5.5,fz;  
    char cx= 'A' ,cy= 'D' ,cz;  
    iz=ix+iy; fz=fx+fy; cz=cx+cy;  
    printf( "%d+%d=%d\n" ,ix,iy,iz);  
    printf( "%.2f+%.2f=%.2f\n" ,fx,fy,fz);  
    printf( "%c+%c=%c\n" ,cx,cy,cz);  
    return 0;  
}
```



• 例：

变量初始化

```
/*An Example*/  
#include <stdio.h>  
int main(void)  
{  
    int ix=3,iy=5,iz;  
    float fx=2.0,fy=5.5,fz;  
    char cx= 'A' ,cy= 'D' ,cz;  
    iz=ix+iy; fz=fx+fy; cz=cx+cy;  
    printf( "%d+%d=%d\n" ,ix,iy,iz);  
    printf( "%.2f+%.2f=%.2f\n" ,fx,fy,fz);  
    printf( "%c+%c=%c\n" ,cx,cy,cz);  
    return 0;  
}
```



• 例：

```
/*An Example*/  
#include <stdio.h>  
int main(void)  
{  
    int ix=3,iy=5,iz;  
    float fx=2.0,fy=5.5,fz;  
    char cx= 'A' ,cy= 'D' ,cz;  
    iz=ix+iy; fz=fx+fy; cz=cx+cy;  
    printf( "%d+%d=%d\n" ,ix,iy,iz);  
    printf( "%.2f+%.2f=%.2f\n" ,fx,fy,fz);  
    printf( "%c+%c=%c\n" ,cx,cy,cz);  
    return 0;  
}
```

变量后使用



定义变量的数据类型的作用

分配合适空间

存储数据

采用的操作不同

数据与数据类型

1. 数据是程序处理的基本对象。根据数据的不同性质和用途可分为不同的数据类型。
2. 不同的数据类型具有不同的存储长度、取值范围、允许的操作。

2.1 基本数据类型

数据类型	关键字	长度	位数	范 围
字符型	char	1	8	-128—127
整 型	int	4	32	-2^{31} — $2^{31}-1$
浮点型	float	4	32	$\pm (10^{-38}-10^{38})$
双精度型	double	8	64	$\pm (10^{-308}-10^{308})$



类型修饰符

short int

long int

long double

unsigned int

unsigned char

signed int

signed char



数据类型

基本类型:

- 整型、字符型、单精度浮点型、双精度浮点型

构造类型:

- 数组、结构、联合、枚举

指针类型