C PROGRAMING

Ketan Kore

Sunbeam Infotech

Type qualifier – const

- const keyword inform compiler that the variable is not intended to be modified.
- Compiler do not allow using any operator on the variable which may modify it e.g. ++, --, =, +=, -=, etc.
- Note that const variables may be modified indirectly using pointers. Compiler only check source code (and do not monitor runtime execution).

Constant pointers

- int a = 10;
- const int *ptr = &a;

- int const *ptr = &a;

- int * const ptr = &a;

- int * ptr const = &a;

- const int * const ptr = &a;
- const int * const ptr = &a;

String

- String is character array terminated with '\0' character.
  - '\0' is character with ASCII value = 0.
- Example :

```
char arr[5] = "abcde";
int j;
for(j=0; j<5; j++)
        printf(“%c”,arr[ j ]);
```

- String input/output
  - char str[20];
  - scanf(“%s”,str); /*Input*/
  - printf(“%s”,str); /*Output*/
  - gets(str); /*Input*/
  - puts(str); /*Output*/
  - scanf("%[^\n]", str); // scan whole line

String functions

- C library have many string functions.
- They are declared in string.h
  - strlen() – size_t strlen(const char *s);
  - strcpy() – char* strcpy(char *dest, const char *src);
  - strcat() – char* strcat(char *dest, const char *src);
  - strcmp() – int strcmp(const char *s1, const char *s2);
  - strchr() – char* strcat(const char *s, int ch);
  - strstr() – char* strstr(const char *s1, const char *s2);
  - strrev() – char* strrev(char *s);

Thank you!

Ketan Kore<ketan.kore@sunbeaminfo.com>