

ソフトウェア 1 シケプリ (2017)

1. ちょっとハマりそうなところ

① Lecture 1

`%f`……double を `printf` など出力する際に使う

`%lf`……double を `scanf` など入力する際に使う

② Lecture 2

`int/int` を double で扱いたい場合、どちらかを double にキャストする

例

```
int n = 1, m = 2;
double x1 = n / m; // 0
double x2 = (double) n / m; // 0.5
```

文字列の最後には必ず `'\0'` が入るので、その分を考えて配列の長さを確保する

③ Lecture 3

`rand()` は、0~`RAND_MAX` の値を返すので、欲しい範囲で割った余りを用いる

例: 100~200 が欲しいとき

```
int random = rand() % 101 + 100;
```

また、呼ぶたびに違う値を返すので、条件分岐などで同じ値を使う場合は変数に代入しておく

`srand()` で `rand()` の初期値を変更できるが、呼ばなくても大丈夫

④ Lecture 4

ポインタは他にシケプリがあったりするので省略

ポインタ変数 = `malloc(typedef(型名) * 要素数);`

可能な限り `free(ポインタ変数);` で開放する

⑤ Lecture 5

`FILE* fp = fopen(ファイル名, アクセス指定子);`

忘れずに `fclose(fp);` をする

`fseek(fp, 0L, SEEK_SET)` // 始めに戻る

`fseek(fp, -1L, SEEK_CUR)` // 1文字戻る

⑥ Lecture 6

`int main(int argc, char *argv[])` としたとき、`argv[0]` にはプログラム名 (`./a.out` など) が入り、実際に指定した引数は `argv[1]` 以降になる

2. ASCII 文字コード

(nul)	0	0000	0x00	(sp)	32	0040	0x20	@	64	0100	0x40	`	96	0140	0x60
(soh)	1	0001	0x01	!	33	0041	0x21	A	65	0101	0x41	a	97	0141	0x61
(stx)	2	0002	0x02	"	34	0042	0x22	B	66	0102	0x42	b	98	0142	0x62
(etx)	3	0003	0x03	#	35	0043	0x23	C	67	0103	0x43	c	99	0143	0x63
(eot)	4	0004	0x04	\$	36	0044	0x24	D	68	0104	0x44	d	100	0144	0x64
(enq)	5	0005	0x05	%	37	0045	0x25	E	69	0105	0x45	e	101	0145	0x65
(ack)	6	0006	0x06	&	38	0046	0x26	F	70	0106	0x46	f	102	0146	0x66
(bel)	7	0007	0x07	'	39	0047	0x27	G	71	0107	0x47	g	103	0147	0x67
(bs)	8	0010	0x08	(40	0050	0x28	H	72	0110	0x48	h	104	0150	0x68
(ht)	9	0011	0x09)	41	0051	0x29	I	73	0111	0x49	i	105	0151	0x69
(nl)	10	0012	0x0a	*	42	0052	0x2a	J	74	0112	0x4a	j	106	0152	0x6a
(vt)	11	0013	0x0b	+	43	0053	0x2b	K	75	0113	0x4b	k	107	0153	0x6b
(np)	12	0014	0x0c	,	44	0054	0x2c	L	76	0114	0x4c	l	108	0154	0x6c
(cr)	13	0015	0x0d	-	45	0055	0x2d	M	77	0115	0x4d	m	109	0155	0x6d
(so)	14	0016	0x0e	.	46	0056	0x2e	N	78	0116	0x4e	n	110	0156	0x6e
(si)	15	0017	0x0f	/	47	0057	0x2f	O	79	0117	0x4f	o	111	0157	0x6f
(dle)	16	0020	0x10	0	48	0060	0x30	P	80	0120	0x50	p	112	0160	0x70
(dc1)	17	0021	0x11	1	49	0061	0x31	Q	81	0121	0x51	q	113	0161	0x71
(dc2)	18	0022	0x12	2	50	0062	0x32	R	82	0122	0x52	r	114	0162	0x72
(dc3)	19	0023	0x13	3	51	0063	0x33	S	83	0123	0x53	s	115	0163	0x73
(dc4)	20	0024	0x14	4	52	0064	0x34	T	84	0124	0x54	t	116	0164	0x74
(nak)	21	0025	0x15	5	53	0065	0x35	U	85	0125	0x55	u	117	0165	0x75
(syn)	22	0026	0x16	6	54	0066	0x36	V	86	0126	0x56	v	118	0166	0x76
(etb)	23	0027	0x17	7	55	0067	0x37	W	87	0127	0x57	w	119	0167	0x77
(can)	24	0030	0x18	8	56	0070	0x38	X	88	0130	0x58	x	120	0170	0x78
(em)	25	0031	0x19	9	57	0071	0x39	Y	89	0131	0x59	y	121	0171	0x79
(sub)	26	0032	0x1a	:	58	0072	0x3a	Z	90	0132	0x5a	z	122	0172	0x7a
(esc)	27	0033	0x1b	;	59	0073	0x3b	[91	0133	0x5b	{	123	0173	0x7b
(fs)	28	0034	0x1c	<	60	0074	0x3c	¥	92	0134	0x5c	 	124	0174	0x7c
(gs)	29	0035	0x1d	=	61	0075	0x3d]	93	0135	0x5d	}	125	0175	0x7d
(rs)	30	0036	0x1e	>	62	0076	0x3e	^	94	0136	0x5e	~	126	0176	0x7e
(us)	31	0037	0x1f	?	63	0077	0x3f	_	95	0137	0x5f	(del)	127	0177	0x7f

A-Z と a-z は連続してないので注意(模擬テストで悩んだ)

3. include ファイル別関数・定数(一部)

詳しくは、テストのページ、1つ目のリンクから「ヘッダファイル別一覧」を参照

① stdio.h

- printf / putchar / puts
- scanf / getc / gets
- fopen / fclose
- getc / putc ……ファイル入出力用なので注意
- perror
- SEEK_SET, SEEK_CUR
- EOF
- NULL

② stdlib.h

- atof / atoi / strtod
- rand / srand
- malloc / free
- abs ……int は stdlib.h
- EXIT_SUCCESS, EXIT_FAILURE
- RAND_MAX

③ math.h

- fabs ……double は math.h
- pow / sqrt

④ string.h

- strcmp
- strlen
- strcpy / strncpy

4. 課題リスト(コピー)(感謝)

① ソフトウェア 1

Lecture 2 【C の基本計算】

課題 2-1 最大、平均、分散の計算

課題 2-2 回分判定

Lecture3 【C の基本構文】

課題 3-1 ナベアツ関数(剰余の活用)

課題 3-2 じゃんけん(乱数の使用)

課題 3-3 標本化関数プロット

課題 3-4 PPAP(文字列配列の操作)

課題 3-5 スロット

課題 3-6 英字のヒストグラム

課題 3-7 英字 2 字組のヒストグラム

課題 3-8 条件付き確率

課題 3-9 アルファベットの情報エントロピー

課題 3-10 条件付き情報エントロピー

Lecture4 【ポインタ】

課題 4-1 ポインタの使用

課題 4-2 ポインタと printf オプション(int)

課題 4-3 ポインタと printf オプション(文字列)

課題 4-4 文字列配列の使用

課題 4-5 Wall コンパイル

課題 4-6 strlen の実装(文字列の字数数え上げ)

課題 4-7 strcpy の実装(文字列のコピー)

課題 4-8 strcmp の実装(文字列の大小比較)

Lecture5 【I/O】

課題 5-1 アニメーション(sleep の使用)

課題 5-2 行列の書き換え

課題 5-3 行列のファイル書き出し

課題 5-4 行列のファイル読み書き

課題 5-5 ライフゲーム

Lecture6 【復讐】

課題 6-1 コマンドライン変数の使用

課題 6-2 モンテカルロ法による円周率計算

課題 6-3 パイコね変換と誤差の拡大

② プログラミング基礎演習

Lecture1

課題 1-1 標準出力とオプション

課題 1-2 for 文を用いた 3 乗根の計算

課題 1-3 ピタゴラス数の列挙

Lecture2 【関数】

課題 2-1 無限級数による円周率の計算

課題 2-2 Zeller の公式による曜日の計算

課題 2-3 素数判定

課題 2-4 オイラー積による円周率の計算

課題 2-5 ユークリッドの互除法

課題 2-6 ピタゴラス数の列挙(互いに素なものの抽出)

課題 2-7 ゴールドバッハ予想の確認

Lecture3 【再帰関数】

課題 3-1 Legendre 多項式の再起計算と計算量

課題 3-2 連分数による平方根の計算

課題 3-3 ニュートン法による方程式の計算

課題 3-4 再起による mod の計算

Lecture4 【デバッグ】

参照するほどのプログラムない…くない？

Lecture5 【ポインタ】

課題 4-1 インクリメント関数

課題 4-2 行列の固有値と固有ベクトル

課題 4-3 文字列の逆転

課題 4-4 文字列の大文字化

課題 4-5 ローマ数字の算用数値化

Lecture6 【I/O】

課題 5-1 データファイルの統計(最大、最小、平均、標準偏差)

課題 5-2 英文のイニシャライズ

課題 5-3 タブ区切りデータの読み込み

5. 過去問答え

① 設問 1

(1):%s

(2):argv[i]

② 設問 2

'b'が表示される.

a[1]は"beta"の文字列の先頭要素へのポインタであり,

*でその先頭要素の中身を表示しているから'b'が表示される.

③ 設問 3

(1):*s

(2):*s

(3):*t

(4):*t

(5):c

④ 設問 4

x,z は setvalue 内のローカル変数に代入され、y だけがグローバル変数に代入されるので、y のみが 5 になる。

⑤ 設問 5

省略

⑥ 設問 6

640120320

⑦ 設問 7

省略