**報告書**

|  |  |
| --- | --- |
| **日付** | 2023－06－23 |
| **報告者** | パク・ジュウォン |

|  |  |  |
| --- | --- | --- |
| アドレスが  高い方向 | **コマンドライン引数と環境変数**  Command -line arguments and environment variables |  |
| **スタック**stack | 大きさは変わる |
| ［注意］  LINUX  ／  GCC  基準 | **自由空間** Free |  |
|  | **ヒープ** Heap | 大きさは変わる |
| アドレスが  低い方向 | **初期化されていないグローバル/静的変数**  BSS(Block Started by Symbol) Segment | 大きさ固定 |
| **初期化されたグローバル/静的変数**  DATA　segment | 大きさ固定 |
| **プログラムコード**  TEXT Segment | 大きさ固定  (読むことしかできない） |

**初期化されたグローバル/静的変数**

・Read WriteDATAsegment

・Read OnlyDATAsegment

：読み取り専用　：char＊str　＝”Hello”；　//str［０］＝’M’；　エラー（修正できない）

コマンド ライン引数と環境変数 : プログラムが開始される環境 (OS)

スタック:地域変数が保存されており、メモリをオペレーティングシステムから受け取ってくる。

あらかじめサイズが分かる変数であるため、コンパイラがより速く動くことができる。

ヒープ:コンパイラがメモリのサイズを事前に不明な場合に使用

**1.配列とruntimeエラー**

連続に繋がってるデータをforを利用して簡単に表せる。

|  |
| --- |
| **1** |

**数字1つ**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | **１** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** |

**数字の配列**

**文字1つ**

|  |
| --- |
| **'a'** |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **'H'** | **'e'** | **'l'** | **'l'** | **'o'** | **'\0'** | **?** | **?** | **?** | **?** |

**文字の配列**

**NULL**

#define MESSAGE "A simbolic string contant"

#define MAXLENGTH 81

int main(){

char words[MAXLENGTH] = "A string in an array";

//最初のアドレスだけを持ってくる。

const char\* pt1 = "A pointer to a string.";  
 //配列は読み取り/書き込みの両方が可能なメモリを使用する。

//文字列リテラルはプログラムの一部であるため、読み取り専用メモリに保存されている。

puts("Puts() adds a newline at the end :");// puts() add \n at the end

put(MESSAGE);

puts(words);//char words[21] removes this warning

puts(pt1);

words[3] = 'p';// 交換 ok

puts(words);

//pt[8] = 'A'; // RunTime Error 読み取り専用メモリに保存されたデータ値を変更しようとすると、オペレーティングシステムが中断されます.

char greeting[50] = "Hello,and""How are" "you" "today...";

// == char greeting[50] = "Hello, and How are you today!";

puts(greeting);

printf("\"to be, or not to be \"hamlet said. \n");

// %p 는 配列のインデックスに入れて初期化する。

// 文字列の前にある逆参照: 最初のスペースのアドレスを逆参照 'e'

printf("%s, %p, %c\n","We","are", \*"excellent programmers");