**報告書**

|  |  |
| --- | --- |
| **日付** | 2023－06－15 |
| **報告者** | パク・ジュウォン |

**1. 変数：メモリの空間を使うために使用する。**

‐どれくらい多くのメモリを保存するか：資料型（int、float、char、等）

‐どこのメモリを使うか

**２.アドレス演算子：＆（ampersand）**

＝他の変数に代入させてくれる。それから一番最初の住所を呼び寄せる。

**アドレスを保存する変数：\*(asterisk）**

**３.注意**

‐大量のデータが保存されているアドレスを直接知る必要がある。  
‐ポインタ変数に保存されている値は、他の変数のアドレスである。  
‐同じ資料型のみ保存できる。

つまり、直接データを保存するよりも、他のメモリ空間やその空間に保存されているデータ(またはオブジェクトobject)を**間接的に指す**（pointer）役割をする。

|  |  |
| --- | --- |
| **주소** | **데이터** |
| **132** | **…** |
| **131** | **8**  **4bytes int** |
| **130** |
| **129** |
| **128** |
| **…** | **…** |
| **107** | **128**  **4 bytes int\*** |
| **106** |
| **105** |
| **104** |
| **103** | **17**  **4bytes int** |
| **102** |
| **101** |
| **100** |
| **099** | **…** |

**・例題**　　　　　　　　　　　　　　　　　　　　　　　　　　　　　**例）**

int a = 7;

Int \*a\_ptr = &a;

\*a\_ptr = 8; 　//8を＊a\_ptrのアドレス値に保存

Int c = 9;

C+= \*a ptr;

**\*(asterisk）**を付けるとアドレスがメモリの空間を使用する。

**\*a\_ptr**は**&a**のアドレス値です

間接アクセスindirection:メモリにアクセスする。  
逆参照dereferencing:データをまたアドレスに変換。  
方向再指定redirection:アドレスを入力した後、アドレス値に現せる。

**・質問**

‐資料型のアドレスの値は全部4byteだと承知していますが、資料型変数はそれぞれのbyteを持っているのにどうして資料型のアドレスの値は皆４byteなのかがまだ理解できていません。