

Nama : Pasha Bhimasty
NIM : L200180123
Kelas : C

TUGAS MODUL 2

1. Kode ASCII (American Standard Code for Information Interchange) adalah suatu standar internasional dalam kode huruf dan simbol seperti Hex dan Unicode tetapi ASCII lebih bersifat universal.

Tabel Kode ASCII :

Karakter	Nilai Binary	Nilai Hexadecimal	Nilai Desimal	Keterangan
!	100001	0021	<u>33</u>	Tanda seru (exclamation)
"	100010	0022	<u>34</u>	Tanda kutip dua
#	100011	0023	<u>35</u>	Tanda pagar (kres)
\$	100100	0024	<u>36</u>	Tanda mata uang dolar
%	100101	0025	<u>37</u>	Tanda persen
&	100110	0026	<u>38</u>	Karakter ampersand (&)
'	100111	0027	<u>39</u>	Karakter Apostrof
(101000	0028	<u>40</u>	Tanda kurung buka
)	101001	0029	<u>41</u>	Tanda kurung tutup
*	101010	002A	<u>42</u>	Karakter asterisk (bintang)
+	101011	002B	<u>43</u>	Tanda tambah (plus)
,	101100	002C	<u>44</u>	Karakter koma
-	101101	002D	<u>45</u>	Karakter hyphen (strip)
.	101110	002E	<u>46</u>	Tanda titik
/	101111	002F	<u>47</u>	Garis miring (<i>slash</i>)
0	110000	0030	<u>48</u>	Angka nol
1	110001	0031	<u>49</u>	Angka satu
2	110010	0032	<u>50</u>	Angka dua
3	110011	0033	<u>51</u>	Angka tiga
4	110100	0034	<u>52</u>	Angka empat
5	110101	0035	<u>53</u>	Angka lima
6	110110	0036	<u>54</u>	Angka enam
7	110111	0037	<u>55</u>	Angka tujuh
8	111000	0038	<u>56</u>	Angka delapan
9	111001	0039	<u>57</u>	Angka sembilan

:	111010	003A	<u>58</u>	Tanda titik dua
;	111011	003B	<u>59</u>	Tanda titik koma
<	111100	003C	<u>60</u>	Tanda lebih kecil
=	111101	003D	<u>61</u>	Tanda sama dengan
>	111110	003E	<u>62</u>	Tanda lebih besar
?	111111	003F	<u>63</u>	Tanda tanya
@	1000000	0040	<u>64</u>	A keong (@)
A	1000001	0041	<u>65</u>	Huruf latin A kapital
B	1000010	0042	<u>66</u>	Huruf latin B kapital
C	1000011	0043	<u>67</u>	Huruf latin C kapital
D	1000100	0044	<u>68</u>	Huruf latin D kapital
E	1000101	0045	<u>69</u>	Huruf latin E kapital
F	1000110	0046	<u>70</u>	Huruf latin F kapital
G	1000111	0047	<u>71</u>	Huruf latin G kapital
H	1001000	0048	<u>72</u>	Huruf latin H kapital
I	1001001	0049	<u>73</u>	Huruf latin I kapital
J	1001010	004A	<u>74</u>	Huruf latin J kapital
K	1001011	004B	<u>75</u>	Huruf latin K kapital
L	1001100	004C	<u>76</u>	Huruf latin L kapital
M	1001101	004D	<u>77</u>	Huruf latin M kapital
N	1001110	004E	<u>78</u>	Huruf latin N kapital
O	1001111	004F	<u>79</u>	Huruf latin O kapital
P	1010000	0050	<u>80</u>	Huruf latin P kapital
Q	1010001	0051	<u>81</u>	Huruf latin Q kapital
R	1010010	0052	<u>82</u>	Huruf latin R kapital
S	1010011	0053	<u>83</u>	Huruf latin S kapital
T	1010100	0054	<u>84</u>	Huruf latin T kapital
U	1010101	0055	<u>85</u>	Huruf latin U kapital
V	1010110	0056	<u>86</u>	Huruf latin V kapital
W	1010111	0057	<u>87</u>	Huruf latin W kapital
X	1011000	0058	<u>88</u>	Huruf latin X kapital
Y	1011001	0059	<u>89</u>	Huruf latin Y kapital
Z	1011010	005A	<u>90</u>	Huruf latin Z kapital

2. Daftar perintah bahasa Assembly

a. MOV

Perintah MOV adalah perintah untuk mengisi, memindahkan, memperbarui isi suatu register, variable ataupun lokasi memory, Adapun tata penulisan perintah MOV adalah : MOV [operand A], [Operand B]

Contoh :

MOV AH,02

Operand A adalah Register AH

Operand B adalah bilangan 02

Hal yang dilakukan oleh komputer untuk perintah diatas adalah memasukan 02 ke register AH.

b. INT (Interrupt)

Bila anda pernah belajar BASIC, maka pasti anda tidak asing lagi dengan perintah GOSUB. Perintah INT juga mempunyai cara kerja yang sama dengan GOSUB, hanya saja subroutine yang dipanggil telah disediakan oleh memory komputer yang terdiri 2 jenis yaitu :

- Bios Interrupt (interrupt yang disediakan oleh BIOS (INT 0 – INT 1F))
- Dos Interrupt (Interrupt yang disediakan oleh DOS (INT 1F – keatas))

c. Push

Adalah perintah untuk memasukan isi register pada stack, dengan tata penulisannya: PUSH [operand 16 bit]

d. Pop

Perintah yang berguna untuk mengeluarkan isi dari register/variable dari stack, dengan tata penulisannya adalah : POP [operand 16 bit]

e. RIP (Register IP)

Perintah ini digunakan untuk memberitahu komputer untuk memulai memproses program dari titik tertentu.

f. A (Assembler)

Perintah Assembler berguna untuk tempat menulis program Assembler.

-A100

0FD8:100

g. RCX (Register CX)

Perintah ini digunakan untuk mengetahui dan memperbarui isi register CX yang merupakan tempat penampungan panjang program yang sedang aktif

Assembly Directive	Keterangan
EQU	Pendefinisian konstanta
DB	Pendefinisian data dengan ukuran satuan 1 byte
DW	Pendefinisian data dengan ukuran satuan 1 word
DBIT	Pendefinisian data dengan ukuran satuan 1 bit
DS	Pemesanan tempat penyimpanan data di RAM
ORG	Inisialisasi alamat mulai program
END	Penanda akhir program
CSEG	Penanda penempatan di code segment
XSEG	Penanda penempatan di external data segment
DSEG	Penanda penempatan di internal direct data

	segment
ISEG	Penanda penempatan di internal indirect data segment
BSEG	Penanda penempatan di bit data segment
CODE	Penanda mulai pendefinisian program
XDATA	Pendefinisian external data
DATA	Pendefinisian internal direct data
IDATA	Pendefinisian internal indirect data
BIT	Pendefinisian data bit
#INCLUDE	Mengikutsertakan file program lain