Differences between compilers and interpreters

|  |  |
| --- | --- |
| INTERPRETER | COMPILER |
| Translates program one statement at a time | Scans the entire program and translates it as a whole |
| It takes less time to analyze the source code but overall execution time is slower | It takes large amount of time to analyze the source code but the overall execution time is comparatively faster |
| No intermediate code is generated. Hence, memory efficient. | Generates intermediate object code which further requires linking. Hence, requires memory |
| Continues translating the program until the first error is met, in which case it stops. Hence, debugging is easy. | It generates the error message only after scanning the whole program. Hence, debugging is comparatively hard. |
| Programming languages like python, ruby use interpreters | Programming languages like C, C++, java use compliers. |

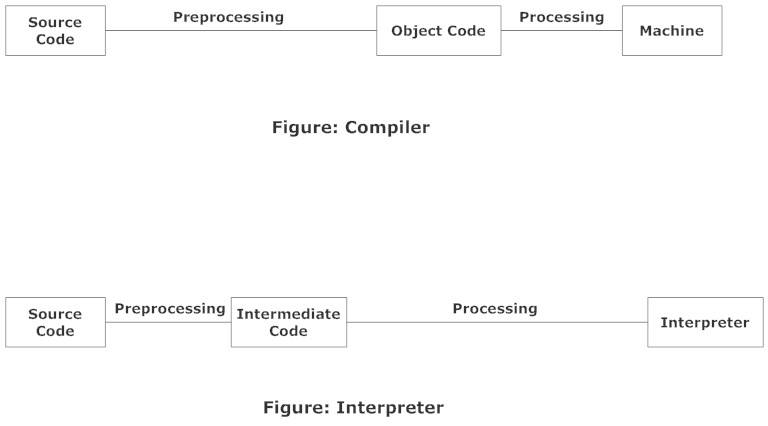


Fig: working of compiler and interpreter