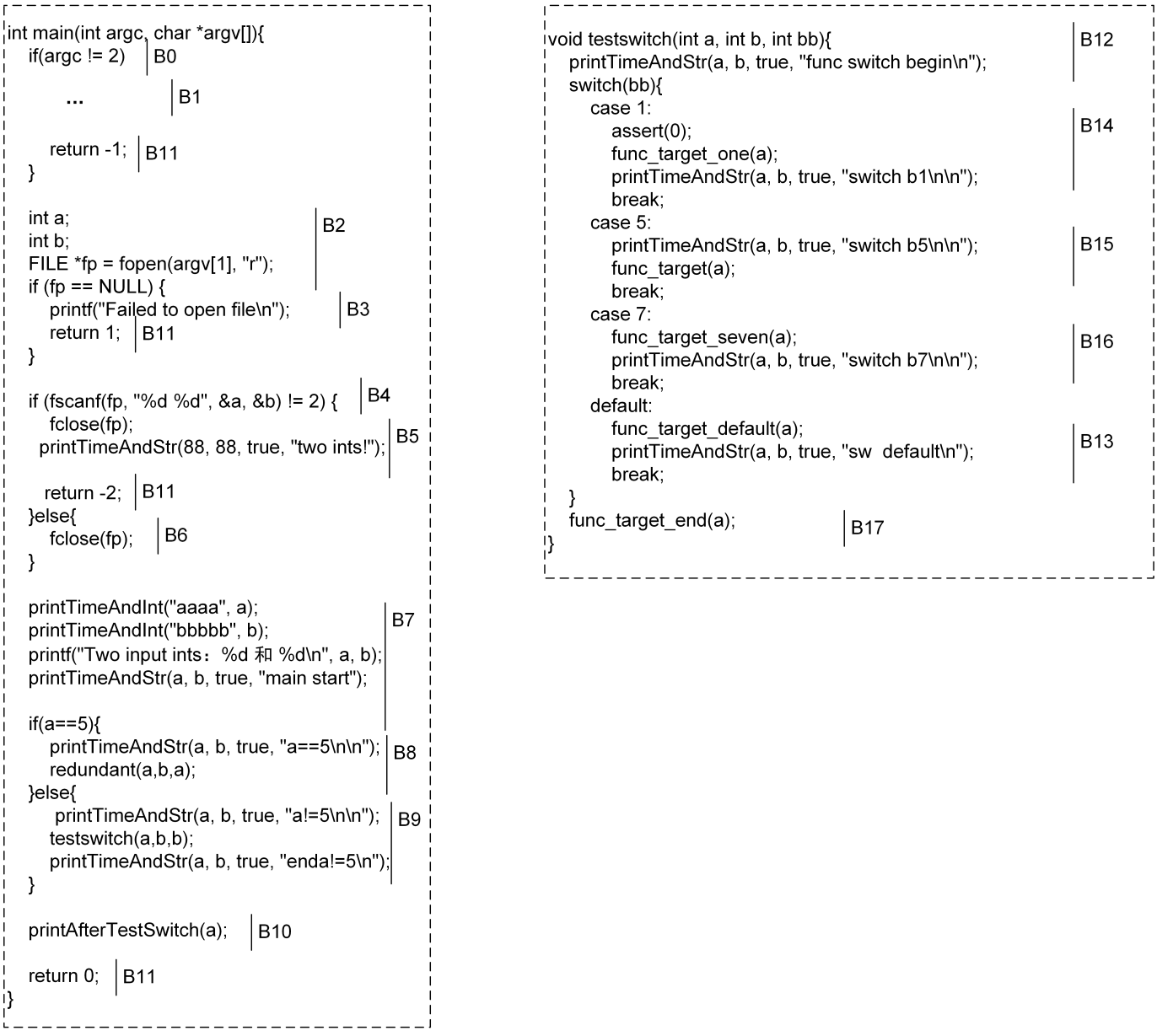

Fig. 1. An example for illustrating the differences between our proposal of redundant basic block termination (rbbt for short), BEACON and SIEVEFUZZ. It contains the source codes with annotations of basic blocks and optimization results of two functions with the above three proposa. Bi (i=0, 1, 2, ..., 17) denotes a basic block. The function *main* is the entry of the program and the function *testswitch* contains the basic block as a target location. × denotes the termination of the execution of a redundant basic block.

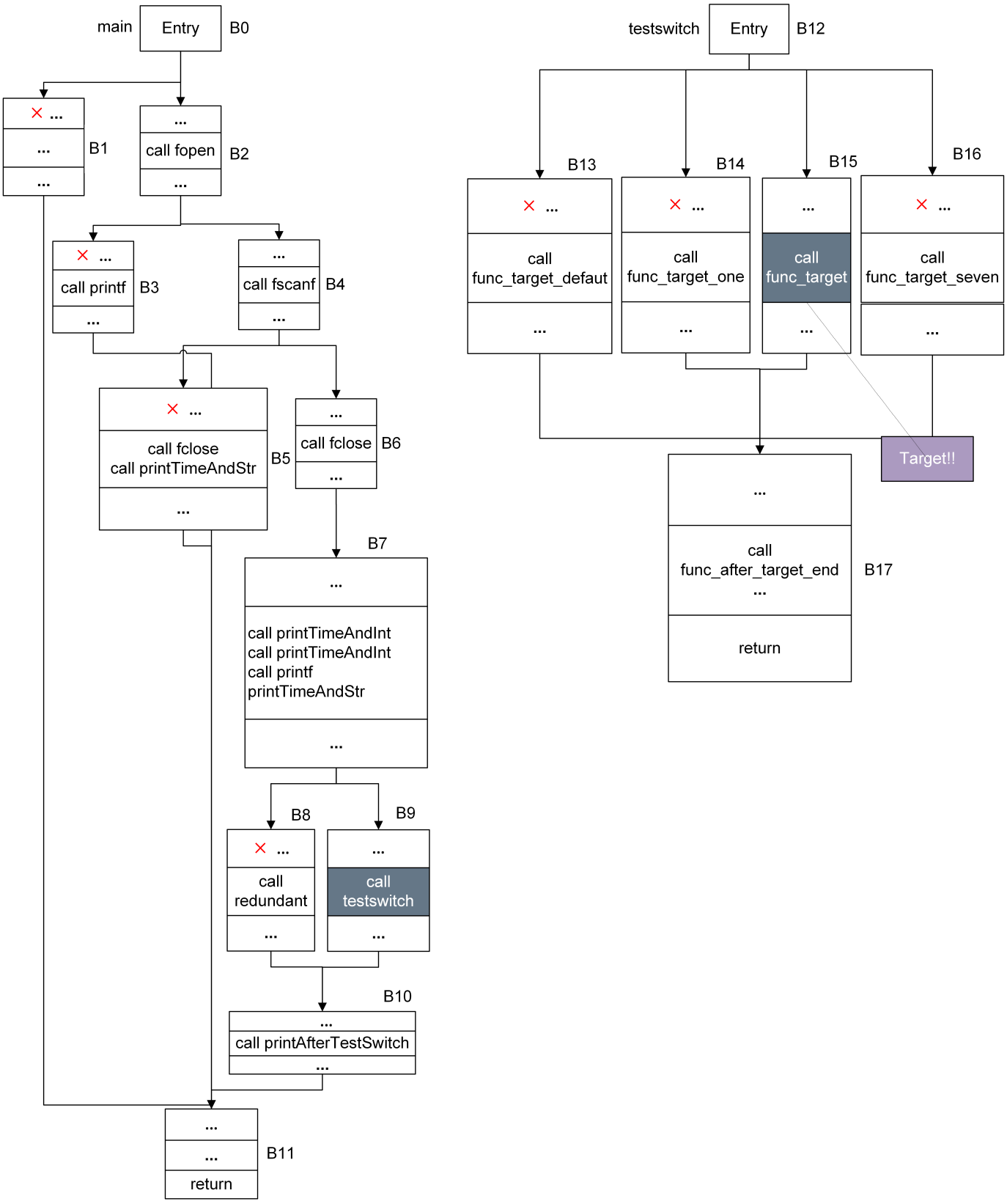

Fig. 2. An example for illustrating our proposal of redundant basic block termination.For the function *testswitch*, its basic blocks of B13, B14, and B16 have no connects to B15 where the target function *func_target* is called. Therefore, these basic blocks would be identified as redundant basic blocks of *testswitch* and would be instrumented with exit instructions to terminate its execution. Similarly, the basic blocks of B1, B3, B5, and B8 would be identified as redundant for *main*. × denotes the termination of the execution of a redundant basic block.

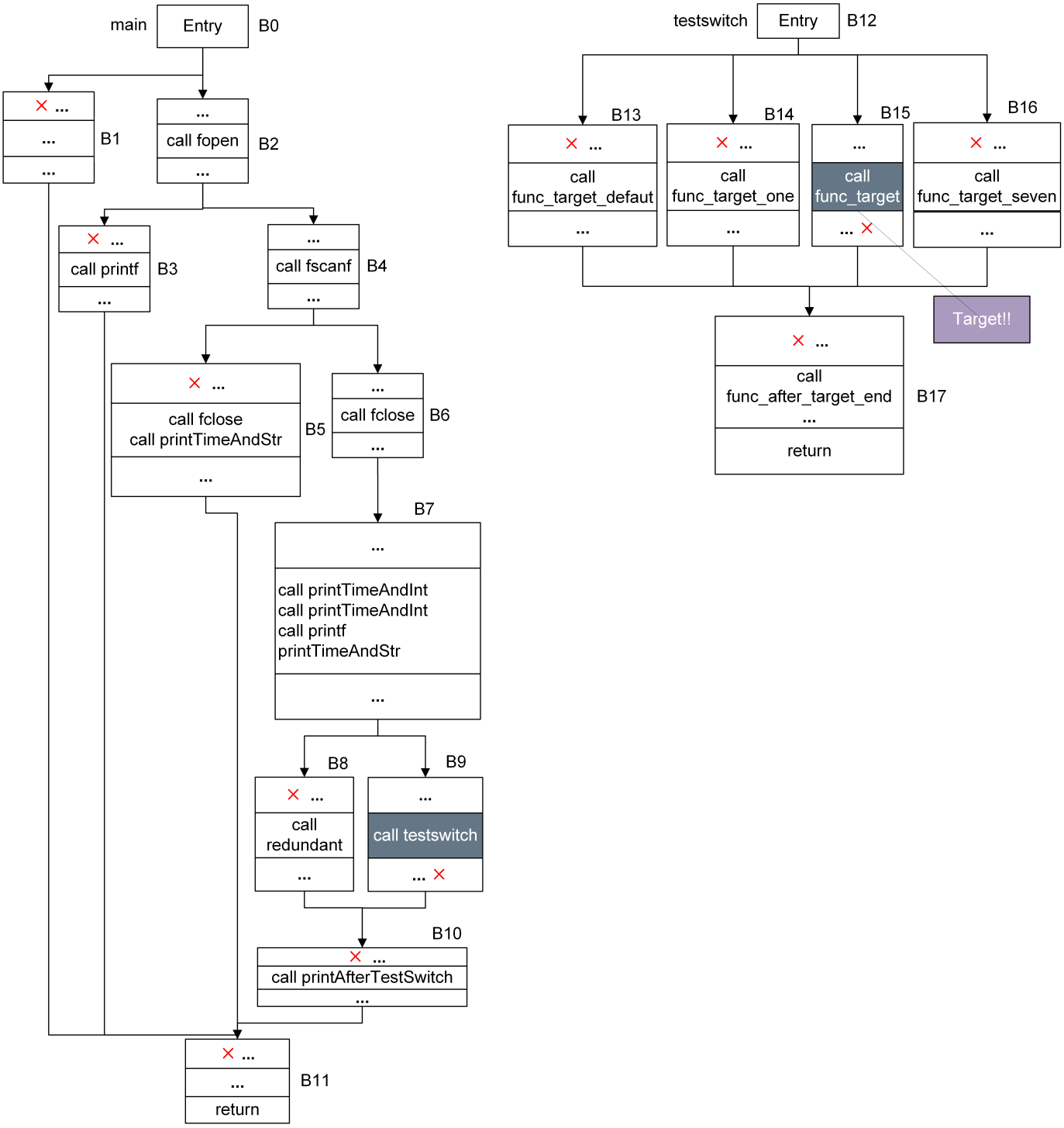

Fig. 3. An example for illustrating BEACON. BEACON reduces the execution of program paths at a basic block-level, which cannot reach the target on the control flow or satisfy the conditional constraints. However, with closed source code, BEACON is provided as an executable binary adapting only for Clang with the version of 4.0, which limits its **generality and availability** for future research. Another difference from our proposal of redundant basic block termination is to **terminate all of the paths after the basic blocks where its parent function calls the target-related functions**. This over-pruning strategy will hindering triggering bugs around the targets. × denotes the termination of the execution of a redundant basic block.

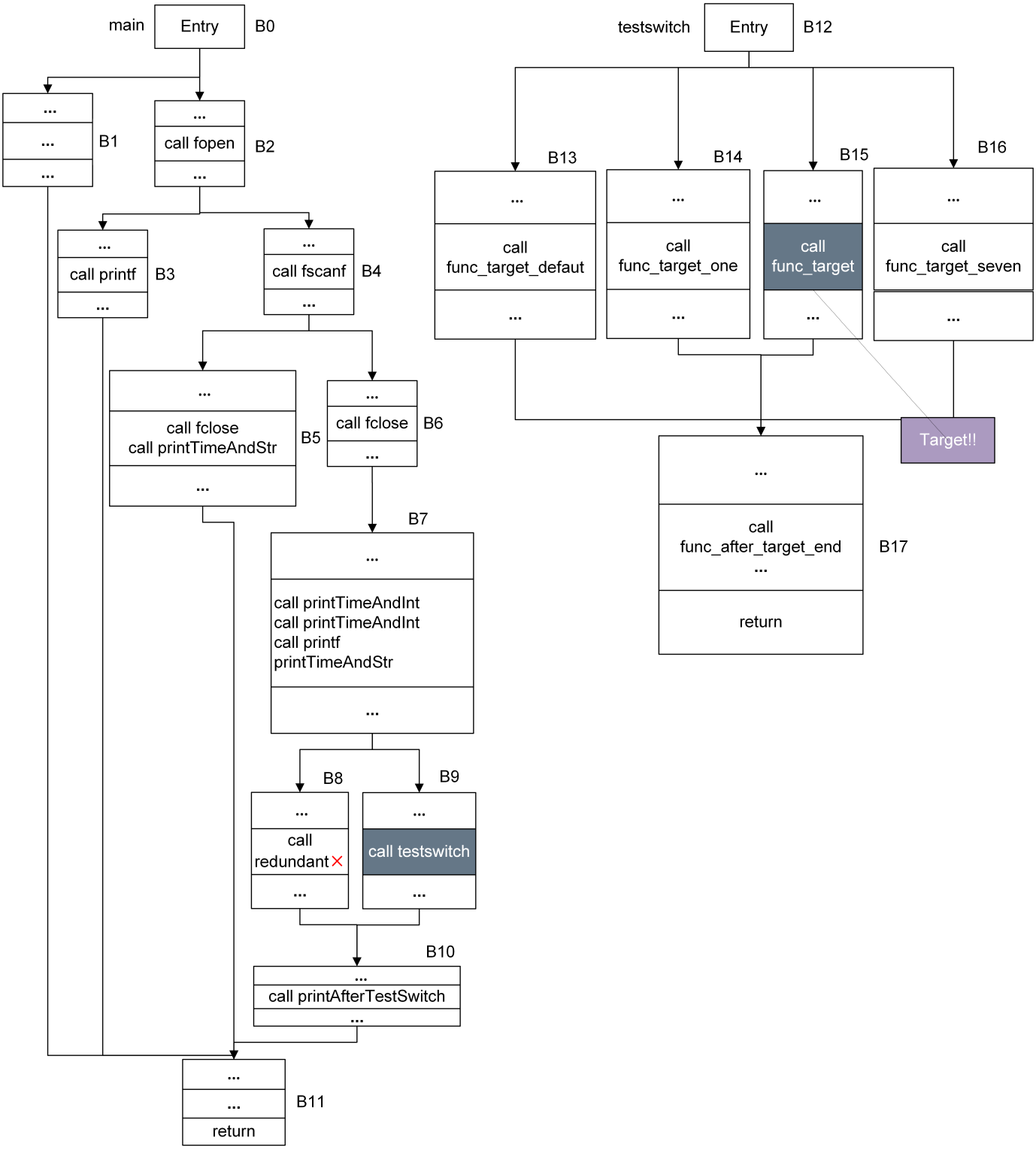

Fig. 4. An example for illustrating SIEVEFUZZ. SIEVEFUZZ preemptively terminates the execution of the functions unrelated to target functions. Because it works at a function level, **it terminates much less basic blocks than our proposal of rbbt**. × denotes the termination of the execution of a target-unrelated function. **Actually it only terminates the execution of the function named redundant in our example**.