Solution to Homework 9 (Finding Invariants, pt 1 & 2)

1. a. Roughly, the invariant will be weaker than the postcondition. (It certainly can't be stronger.)
   b. The invariant must be true every time control is at the ***while*** test, including the first time.
   c. The initialization code must establish the invariant even if we know we'll do zero iterations.
   d. We can have $\neg B$ true anywhere inside the loop body (just not at its beginning or end).

2. $u := 0;$ {***inv*** $(x^2 - f(2*y, u) < g(z^2, b) \wedge 0 \le u \le n$} ***while*** $u \ne a$ *do …;* $u := u+1$ ***od***
   $v := -1;$ {***inv*** $(x^2 - f(2*y, a) < g(z^2, v) \wedge -n \le v \le -1$} ***while*** $v \ne b$ *do …;* $v := v-1$ ***od***
   Replacing 2 by w in $2*y$ gives a candidate invariant of $(x^2 - f(w*y, a) < g(z^2, v)$ but we don't know enough about the range of w to initialize it or to write a progress step.

3. For the postcondition $(x > 0 \vee y < n) \wedge (x < n \rightarrow f(x, n)) \wedge (f(y, n) \leftrightarrow y \ge 0)$, the invariants if we drop a conjunction are:
   a. {***inv*** $(x < n \rightarrow f(x, n)) \wedge (f(y, n) \leftrightarrow y \ge 0)$} ***while*** $x \le 0 \wedge y \ge n$ …
   b. {***inv*** $(x > 0 \vee y < n) \wedge (f(y, n) \leftrightarrow y \ge 0)$} ***while*** $x < n \wedge \neg f(x, n)$
   c. {***inv*** $(x > 0 \vee y < n) \wedge (x < n \rightarrow f(x, n))$} ***while*** $f(y, n) \oplus y \ge 0$ (where $\oplus$ is logical XOR)

4. (Add a disjunct)
   a. Taking the postcondition $p_1 \wedge p_2$ and dropping p_1 is the same as adding the disjunct $\neg p_1 \wedge p_2$ to $p_1 \wedge p_2$. Similarly, dropping p_2 is the same as adding $(p_1 \wedge \neg p_2)$ as a disjunct
   b. *Add a Disjunct* is less constrained than *Replace a Constant by a Variable* or *Drop a Conjunct* because we can add any predicate as the disjunct (so long as we can test it). Replacing a constant by a variable is constrained by what constants appear in the postcondition. Dropping a conjunct is constrained by the number of conjuncts available.

5. (Full outline for Example 6: Faster Multiplication)
   As in Example 5, x is the bound function. There's a slight complication in that after the ***if*** $odd(x)$ … statement, either $x = x_0$ or $x_0 - 1$. For simplicity, I decided to use $x \le x_0$ instead.

   $\{x = x_0 \wedge y = y_0 \wedge x_0 \ge 0\}$
   $z := 0;$
   $\{x = x_0 \wedge y = y_0 \wedge x_0 \ge 0 \wedge z = 0\}$
   {***inv*** $p \equiv z = x_0*y_0 - x*y \wedge x \ge 0$} {***bd*** x}
   while $x \ne 0$ ***do***
   $\quad \{p \wedge x \ne 0 \wedge x = x_0\}$

if odd(x) ***then***

    $\{p \wedge x \neq 0 \wedge odd(x) \wedge x = x_0\}$

    $\{p[x-1/x][z+y/z] \wedge even(x-1) \wedge x_0 \neq 0 \wedge 0 \leq x-1 \leq x_0\}$

    z := z+y;

    $\{p[x-1/x] \wedge even(x-1) \wedge x_0 \neq 0 \wedge 0 \leq x-1 \leq x_0\}$

    x := x−1

    $\{p \wedge even(x) \wedge x_0 \neq 0 \wedge 0 \leq x \leq x_0\}$

else

    $\{p \wedge x = x_0 \wedge x \neq 0 \wedge \neg odd(x)\}$

    skip

    $\{p \wedge x = x_0 \wedge x \neq 0 \wedge \neg odd(x)\}$

    $\{p \wedge even(x) \wedge x_0 \neq 0 \wedge 0 \leq x \leq x_0\}$

fi;

$\{p \wedge even(x) \wedge x_0 \neq 0 \wedge 0 \leq x \leq x_0\}$

$\{p[x \div 2/x][2*y/y] \wedge x \div 2 < x_0\}$

y := 2*y;

$\{p[x \div 2/x] \wedge x \div 2 < x_0\}$

x := x÷2

$\{p \wedge x < x_0\}$

od

$\{(p \equiv z = x_0*y_0 - x*y) \wedge x = 0\}$ $\{z = x_0*y_0\}$