Given a finite dimensional real vector space \( V \) with a real symmetric bilinear form \( \langle \ , \ \rangle \), here are algorithms for finding a basis for the space which is orthogonal with respect to \( \langle \ , \ \rangle \).

**Positive-definite case:** The Gram-Schmidt algorithm applies when \( \langle \ , \ \rangle \) is positive-definite. Suppose \( S \) is a finite set that spans a subspace \( V \) of a real vector space \( W \) with a positive definite real symmetric form \( \langle \ , \ \rangle \). Last semester I gave a version of the Gram-Schmidt algorithm that can be used to obtain from \( S \) an orthogonal basis \( B \) for \( V \). (The algorithm in Artin’s book assumes \( S \) is a basis of \( V \) and that \( W = V \).) To make it easier to see how to generalize Gram-Schmidt to handle non-positive definite forms, I’ll describe Gram-Schmidt using slightly different notation than last semester. If \( S = \{s_1, \ldots, s_r\} \), the procedure from last semester gives rise to vectors \( \{u_1, \ldots, u_r\} \), and then \( B = \{u_i : u_i \neq 0\} \).

The set \( B \) you end up with is a basis for \( \text{Span}(\{s_1, \ldots, s_r\}) \) orthogonal with respect to \( \langle \ , \ \rangle \), but not necessarily orthonormal. You need to divide each \( u \) in \( B \) by \( \sqrt{c_u} \) to get an orthonormal basis.

**A general case algorithm:** In general (i.e., if \( \langle \ , \ \rangle \) is not necessarily positive definite), a modified version of Gram-Schmidt can be used. Here are the steps:

1. Pick a basis \( B \) for the nullspace \( N \) of \( \langle \ , \ \rangle \). (So at the start of this algorithm \( B \) is not empty.)

2. Extend \( B \) to a basis \( B \cup S \) of \( V \). Let \( S' = \emptyset \).

3. Let \( v \) be the first element of \( S \), and take it out of \( S \) (i.e., redefine \( S \) to be \( S - \{v\} \)). Define \( v' \) to be

\[
v' = v - \sum_{u \in B} \langle v, u \rangle u / c_u
\]

and let \( c_{v'} = \langle v', v' \rangle \). If \( c_{v'} = 0 \), then add \( v' \) to \( B \) (i.e., redefine \( B \) to be \( B \cup \{v'\} \)). Otherwise leave \( B \) alone.
(4) Keep repeating step 3. Eventually, either $S$ and $S'$ will both be empty (in which case you’re done, and $B$ is a basis for $V$ orthogonal with respect to $\langle \ , \ \rangle$), or $S$ will be empty but $S'$ will not be. In this second situation, every element of $S'$ is orthogonal to every element of $B$, but $c_x = 0$ for every $x \in S'$. If $S'$ has only a single element, move it into $B$ (i.e., redefine $B$ to be $B \cup S'$); then $B$ is an orthogonal basis and you’re done. If $S'$ has two or more elements, let $u = v + w$, where $v$ is the first element of $S'$ and $w$ is any element of $S'$ such that $\langle v, w \rangle \neq 0$ (see the note (*) for why such a $w$ exists), then move $u$ into $B$, move all but the first element $v$ of $S'$ back into $S$, reset $S'$ to be empty, and go back to repeating step 3 (and step 4 if it ever happens that $S$ becomes empty but $S'$ is not). The end result of applying steps 3 and 4 is that $S$ gets smaller and $S'$ is empty, so eventually both $S$ and $S'$ will be empty, and you’re done: the $B$ you end up with is an orthogonal basis for $V$. (Note *: $c_u = \langle u, u \rangle$ simplifies to $2\langle v, w \rangle$, hence is nonzero. But how do we know that an appropriate $w$ exists? Recall $N$ is in the span of $B$ and $B \cup S' \cup S$ is always a basis for $V$, so no element of $S'$ can be in $N$. If no $w$ exists, this means $v$ is orthogonal to every element of $S'$, but $v$, being in $S'$, is also orthogonal to every element of $B$, which would mean $v$ is in $N$.)

The nonexplicit algorithm from last semester for the general case: I gave a somewhat nonexplicit routine for the general case last semester. It’s similar to but conceptually a bit simpler than the general case algorithm above; however, it’s not as efficient. The main difference is that in the algorithm above we try to choose elements of $S$ one at a time, adjusting each choice to make it orthogonal to what is already in $B$, then (as long as $c \neq 0$ for our adjusted choice) we add it to $B$ and delete our choice from $S$. Only when $c = 0$ for every choice do we do something different, which involves extra work. The algorithm from last semester does this extra work every time an element is to be included in $B$. Assume $\operatorname{dim} V = n$ and let $N$ be the nullspace of $\langle \ , \ \rangle$.

(1) If $n = 1$ or $N = V$, any basis $B$ is orthogonal with respect to $\langle \ , \ \rangle$.

(2) If $n > 1$ and $N$ is a proper subspace of $V$, then:

(i) pick $w \in V$ such that $\langle w, w \rangle \neq 0$; such a $w$ exists by Proposition 2.2 on p. 243. Let $W = \operatorname{Span}(\{w\})$ and note that $\operatorname{dim} W^\perp = n - 1$.

(ii) pick a basis $B'$ of $W^\perp$ orthogonal with respect to $\langle \ , \ \rangle$.

(3) Then $B = B' \cup \{w\}$ is a basis of $V$ which is orthogonal with respect to $\langle \ , \ \rangle$. Note that (2)(ii) is iterative: if either $\operatorname{dim} W^\perp = 1$ or the nullspace of $W^\perp$ is all of $W^\perp$, then (as in (1)) any basis $B'$ of $W^\perp$ is orthogonal. If neither condition obtains, we repeat step (2) (i.e., we pick a new vector $w$, this time in $W^\perp$, etc) and get a new (and smaller) $W^\perp$. Because the dimension of $W^\perp$ keeps getting smaller, eventually step (1) will apply; our sequence of choices of $w$ together with any basis of the final $W^\perp$ gives an orthogonal basis for $V$. 